AD/A-003 176

A USER'S MANUAL FOR THE REPSIL CODE

J. M. Santiago

Ballistic Research Laboratories

Prepared for:

Army Materiel Command

October 1974

DISTRIBUTED BY:

NTIS

National Technical Information Service
U. S. DEPARTMENT OF COMMERCE

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER BRL REPORT NO. 1744	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER AD/A 003176
4. TITLE (and Subtitle) A USER'S MANUAL FOR THE REPSIL CODE		5. TYPE OF REPORT & PERIOD COVERED Final
		6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) J.M. Santiago H.L. Wisniewski N. J. Huffington, Jr.		8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS USA Ballistic Research Laboratories Aberdeen Proving Ground, Maryland 21005		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS RDT&E 1W162118AD51 RDT&E 1T161102A33E
11. CONTROLLING OFFICE NAME AND ADDRESS U.S. Army Materiel Command 5001 Eisenhower Avenue Alexandria, Virginia 22304		12. REPORT DATE OCTOBER 1974
		13. NUMBER OF PAGES 204
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)		15. SECURITY CLASS. (of this report) UNCLASSIFIED
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Finite Difference Method
Computer Program
Large Deflections
Thin Kirchhoff Shells
Elastic-Plastic Behavior
Impulsive/Pressure Loadings

20. ABSTRACT (Continue on reverse side if necessary and identify by block number) (njs) & (bj)

REPSIL is a finite difference computer program which calculates the large deflection, transient motion of thin Kirchhoff shells. The program marches out the solution by cyclically solving an explicit, centered time difference formula for displacements. The formulation handles elastic-plastic behavior, including strain hardening and strain rate effects, and permits the modeling of a variety of initial shell configurations and impulsive/pressure loadings.

(OVER)

DD FORM 1 JAN 73 1473

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

Item 20 (Continued) ABSTRACT

The manual gives instructions for correctly setting up problems and estimating machine time and storage requirements. Two illustrative problems are set up and the resulting solutions given. The numerical algorithm employed by REPSIL is outlined and instructions for programming additional initial geometries and loadings are given. The REPSIL plotting program, which produces isometric and cross-sectional displays and time histories of energies, deflections and strains, is also described. Listings of both the REPSIL program and the REPSIL plotting program are included.

TABLE OF CONTENTS

TABLE OF CONTENTS (Cont'd)

TABLE OF CONTENTS (Cont'd)

LIST OF ILLUSTRATIONS

LIST OF ILLUSTRATIONS (Cont'd)

LIST OF TABLES

LIST OF TABLES (Cont'd)

LIST OF SYMBOLS

		Page
a	Determinant of surface metric $a_{\alpha\beta}$.	34
a_o	Initial value of determinant, a, at time t = 0.	28
$a_{\alpha\beta}$	Covariant components of middle surface metric	34
$a^{\alpha\beta}$	Contravariant components of middle surface metric	34
A	Coefficient in quadratic equation for $\Delta\lambda$, see (2.26)	38
$A_{\alpha\beta}$	Initial values of $a_{\alpha\beta}$ at strain locations on bounding surfaces of shell	52
$b_{\alpha\beta}$	Covariant components of 2nd fundamental tensor of middle surface	34
b^{α}_{β}	Mixed components of 2nd fundamental tensor of middle surface.	34
B	Coefficient in quadratic equation for $\Delta\lambda$, see (2.26)	38
$B_{\alpha\beta}$	Initial values of $b_{\alpha\beta}$ at strain locations on bounding surfaces of shell	52
c	Speed of longitudinal waves in plate	24
C	Coefficient in quadratic equation for $\Delta\lambda$, see (2.26)	38
d, d_j	Empirical constants used to model strain rate dependence	37,63
D	Coefficient of viscous damping	42
E	Young's modulus.	24
E_1, E_2	Exact elongational surface strains along η^1 and η^2 coordinate directions.	52

LIST OF SYMBOLS (Cont'd)

		Page
E_θ	Exact elongational surface strain along direction making angle θ with η^1 coordinate curve	53
g	Determinant of surface metric $g_{\alpha\beta}$.	35
$g_{\alpha\beta}$	Covariant components of metric for surface ζ distant from middle surface.	35
$g^{\alpha\beta}$	Contravariant components of metric for surface ζ distant from middle surface.	35
$G_{\alpha\beta}$	Initial values of $g_{\alpha\beta}$ at strain locations on bounding surfaces of shell	52
h	Thickness dimension of shell	22
k	Integer specifying layer station, see (2.2).	22
K	Number layers into which shell thickness is divided.	22
ℓ	Number of time steps from initial to current time	30
L	Number of subincrements into which elastic stress increment is divided for the plasticity calculations.	131
m	Integer specifying mesh point along η^1 coordinate curve, also see (6.1) and (6.2).	22
$M^{\alpha\beta}$	Components of stress moment resultant, called the bending resultant tensor	39
n	Integer specifying mesh point along η^2 coordinate curve, also see (6.1) and (6.2).	22
n^i	Cartesian components of unit normal to middle surface at current time t, also see (2.11)	32
n_-^i	Values of N^i components at time t_-	30

LIST OF SYMBOLS (Cont'd)

Symbol	Description	Page
$N^{i\alpha}$	Cartesian components of stress resultant tensor	39
p, p_j	Empirical constants used to model strain rate dependence	37,63
P	Pressure acting on shell	32
P^*	Force due to pressure acting on unit element, $\Delta\eta^1\Delta\eta^2$, of material coordinate area, called the augmented pressure.	34
$Q^{\alpha\beta}$	Membrane or tangential components of the stress resultant tensor.	39
t	Current time; $t = \ell\Delta t$.	21
t_-	Time at one increment before current time; $t = (\ell-1)\Delta t$.	30
T	Kinetic energy of shell at current time t. . .	45
T_-	Kinetic energy of shell at time $t - \frac{1}{2}\Delta t$ . . .	48
T_+, T_{++}	Kinetic energy of shell at time $t + \frac{1}{2}\Delta t$ and $t + \frac{3}{2}\Delta t$	48,49
T^*	Kinetic energy removed by previous KEA operation.	49
u^i_α	First partial derivatives of Δu^i with respect to η^α.	32
$u^i_{\alpha\beta}$	Second partial derivatives of Δu^i with respect to η^α.	32
U^i	Total current displacement at initially specified location	53
v	Magnitude of initial normal velocity imparted to shell	28

LIST OF SYMBOLS (Cont'd)

		Page
v^i	Cartesian components of initial normal velocity imparted to shell	28
V	Elastic strain energy	44
W	Total work performed by pressure on shell up to current time	46
W_D	Energy dissipated by damping and KEA procedure, called the damping work	48
W_P	Energy dissipated by plastic flow, called the plastic work	46
y^i	Cartesian coordinates of middle surface at current time t	32
y_-^i	Cartesian coordinates of middle surface at time t_-	30
y_α^i	First partial derivatives of y^i with respect to η^α	32
$y_{\alpha\beta}^i$	Second partial derivatives of y^i with respect to η^α	32
α	Coefficient, function of θ, used in computing surface strains	52
β	Coefficient, function of θ, used in computing surface strains	52
γ	Shear component of tangential surface strain	52
Γ_o	Mass per unit initial middle surface area; $\Gamma_c = \rho h$	28
$\Gamma_{\alpha\beta}{}^\gamma$	Christoffel symbol based on middle surface metric, $a_{\alpha\beta}$	34
δ	Coefficient, function of $G_{\alpha\beta}$, used in computing surface strains	52

LIST OF SYMBOLS (Cont'd)

Symbol	Description	Page
$\dot{\varepsilon}$	Second invariant of the strain rate deviator	37
$\varepsilon_1, \varepsilon_2$	Elongational components of tangential surface strain	52
$\varepsilon_j, \varepsilon_j^o (j=1,2,\ldots)$	Strain coordinates on polygonal approximation to strain hardening stress-strain curve where slopes change	63
$\varepsilon_{\alpha\beta}$	Covariant components of tangential strain	50
ζ	Normal distance of particles from middle surface	21
η^α	Material coordinate of particles on middle surface	21
θ	Angle specifying direction of elongational surface strain relative to η^1 coordinate curve, see Figure 3.5	50
ν	Poisson's ratio	24
ρ	Initial mass density	24
σ_o	Yield stress	37
$\sigma_j, \sigma_j^o (j=1,2,\ldots)$	Stress coordinates on polygonal approximation to strain hardening stress-strain curve where slopes change	63
$\sigma^{\alpha\beta}$	Contravariant components of tangential stress at time t	32
$\sigma_-^{\alpha\beta}$	Contravariant components of tangential stress at time t_-	30
$\overset{P}{\sigma}{}^{\alpha}_{\beta}$	Components of plastic flow corrector stress	37
$\overset{T}{\sigma}{}^{\alpha}_{\beta}$	Components of trial stress (assuming stress increment to be elastic)	37

LIST OF SYMBOLS (Cont'd)

Symbol	Description	Page
σ^{α}_{β}	Mixed components of tangential stress at time t	38
$\sigma_{-}{}^{\alpha}_{\beta}$	Mixed components of tangential stress at time t_-.	35
ϕ	Function specifying von Mises yield surface. .	37
$\Delta a_{\alpha\beta}$	Increment in $a_{\alpha\beta}$ during interval Δt.	35
$\Delta b_{\alpha\beta}$	Increment in $b_{\alpha\beta}$ during interval Δt.	35
Δn	Normal component of incremental change in n^i during interval Δt	35
Δn_{α}	Tangential components of incremental change in n^i during interval Δt.	35
Δt	Finite difference time increment	23
Δt^*	Diminish time increment to prevent instability during KEA procedure	49
Δt_B	Maximum stable time increment for membrane motion of plate.	24
Δt_M	Maximum stable time increment for bending motion of plate.	24
ΔU_n	Correction applied to normal component of Δu^i_+ one mesh spacing in from clamped boundary	43
Δu^i	Cartesian components of displacement increment undergone by middle surface during interval $\Delta t = t - t_-$.	30
Δu^i_+	Value of Δu^i for time interval following current time t	32
Δu^i_*	Value of Δu^i_+ one mesh spacing in from clamped boundary before application of normal correction Δu_n	43

LIST OF SYMBOLS (Cont'd)

1. INTRODUCTION

REPSIL* is a FORTRAN IV computer program developed at the BRL to treat the large transient deformations of anelastic shells under blast loadings. The program uses the finite difference technique to solve the equations governing the motion of thin Kirchhoff shells of negligible rotational inertia. These equations are derived in a recent BRL report [1]**. That report, which also treats more general Kirchhoff shells, constitutes the theoretical documentation for REPSIL. The present report is a companion user's manual.

The equations on which REPSIL is based impose certain restrictions on the types of shells and deformations that can be treated. As already mentioned, only thin Kirchhoff shells of negligible rotational inertia can be treated. Moreover, as presently formulated, the program can only handle shells of uniform thickness having no cutouts, stiffeners, or bifurcations. On the other hand, the formulation does treat the geometric nonlinearities due to large displacements exactly.

Within the limits of the above geometric restrictions, REPSIL can accept a wide variety of initial shell geometries. This is made possible by the finite different formulation being coded independent of any particular initial geometry. Hence, accommodating any initial geometry simply entails programming it into the initial geometry subroutine according to the instructions contained in Chapter 6. The current version of REPSIL comes with initial geometry subroutines for the rectangular plate, the cylindrical shell and the conical shell; Section 2.3 contains instructions for their use.

The program also can accept arbitrary distributions over the shell surface of initial impulse velocities and time varying pressures. The minor programming necessary to specify such distributions is described in Chapter 6. Presently, initial velocities at mesh points and over rectangular regions of the shell surface can be specified on input cards without need of programming, as detailed in Section 3.2.

REPSIL can simulate a number of boundary conditions along the four edges bounding the shell. Three edges can have any combination of clamped or hinged boundary conditions. The remaining edge and two of the previous edges can be edges constrained to move in symmetry planes of the problem. The boundary conditions are described in detail in Section 3.2, where the procedure for selecting them is given.

---

* Response of Elastic-Plastic Shells to Impulsive Loadings

** Numbers in brackets correspond to the list of references on page 126.

Preceding page blank

REPSIL can model a variety of anelastic material responses, although the present version is limited to isotropic materials. Within that limitation, the response can be either completely elastic or elastoplastic. In the elastic range the behavior is linear. The plastic response can be perfect or strain hardening and in either case it can be made strain rate dependent. Section 3.2 gives the precedure for eliciting these material properties.

A useful feature of the code is the automatic determination of an optimum, stable time increment. REPSIL uses an explicit finite difference approximation to the equations of motion; as is well known, such explicit schemes are subject to numerical instability unless the size of the time increment is limited to some maximum increment. The REPSIL program initially computes such a maximum increment based on criteria given in Section 2.2 and then uses this increment or the increment specified by the initial data (see Section 3.2), whichever is smaller, for the (constant) time increment used to generate the solution. Hence, a stable solution is guaranteed.

The program is provided with a damping option that permits the rapid attainment of final deformed configurations by numerically damping out the motion of the shell. Damping is mainly achieved through the artifice of automatically annihilating the kinetic energy whenever it reaches a local maximum. Details on this procedure are given in Section 2.4, while instructions for executing the option are in Section 3.2.

REPSIL comes equipped with a number of print options for outputting results of the calculations at regular intervals. Printed out are such local measures of the deformation as the displacements and surface strains, and such global measures as the kinetic energy, strain energy and plastic work. There is also a companion plotting program that at regular intervals draws isometric and cross-sectional views of the deforming shell and that graphs time histories of the printed measures of deformations just mentioned. The output options are described in Chapter 4 and illustrated by example problems in Chapter 5. Appendix D contains a description and a listing of the plotting program.

This report is organized as follows: Chapter 2 presents the assumptions and equations of the REPSIL formulation and outlines the computational algorithm used to solve the equations; Chapter 3 describes the input data, shows how to initiate and continue a problem and select the various options, and gives rules-of-thumb for estimating the memory and machine time requirements for a problem; Chapter 4 describes the various output data generated by REPSIL and the formats in which they are printed; Chapter 5 gives two example problems, describing how they are set up and the kind of output data generated; Chapter 6 gives instructions for programming arbitrary loadings and initial geometries. Chapter 3, 4 and 5 contain the necessary information to run the program as listed in Appendix E, with the associated loading and initial geometry subroutines. A user desiring to model other loadings or initial geometries

should read Chapter 6 for programming instructions. Such a user would find a reading of Chapter 2 useful as background. On the other hand, the analyst or programmer who intends to make extensive changes in the formulation or numerics of the program would do well to thoroughly study Chapter 2 before proceeding to the study of Appendix E with the aid of Appendix C, the list of program variables.

2. DESCRIPTION OF PROGRAM

2.1 Synopsis of Program

REPSIL treats the transient deformation of shells by approximating the initial value problem with an explicit nonlinear finite difference scheme. The finite difference equations are solved at each time increment to update the deformation variables. This procedure is repeated cyclically in order to generate the time history of the deformation. This chapter presents the equations of the REPSIL formulation and describes the computational algorithm by which REPSIL solves them.

The formulation is based on equations that use a material or Lagrangian description: the dependent variables of the theory are functions of η^α *, the material coordinates of the middle surface of the shell; moreover variables defined outside the middle surface also depend upon ζ, the normal distance from the middle surface **. All variables are functions of t, the time.

Some comments on the physical significance of the material coordinates of the middle surface are in order. For simple initial geometries, the middle surface material coordinates often have significance as distances, arc lengths, or angles ***; however, this need not be the case, not even for simple geometries. It is better to regard the material coordinates abstractly as a pair of parameters defining the position of the middle surface in space. In other words, the material coordinates serve as parameters for the parametric representation of the middle surface in space. That the parameters η^α are also material coordinates simply means that as the image of the middle surface changes in space as a function of time, the pair of values η^α associated with a given material particle on the middle surface does not change, but remains fixed.

---

* Index notation is used, with Greek indices ranging over the integers 1 and 2; hence, $\eta^\alpha \equiv \eta^1 , \eta^2$

** The concept of material coordinates for a shell is defined more precisely in [1].

*** For example, see the cases of the flat plate, cylindrical shell and conical shell in Section 3.2.

The finite difference scheme is obtained by making the dependent variables of the theory discrete functions of η^1, η^2, ζ, t. Discretizing the dependence on η^α makes the dependent variables functions of the intersection points of the two dimensional rectangular mesh resulting from the division of the domain of η^α into increments (see Figure 3.1 and 6.1); more precisely, the variables become functions of the mesh number pair (m,n), the integer pair corresponding to the mesh intersection whose coordinates η^α satisfy the relations

$$\eta^1 = m\,\Delta\eta^1 + \eta_o^{\,1} \quad , \quad \eta^2 = n\,\Delta\eta^2 + \eta_o^{\,2} \,, \qquad (2.1)$$

with $\Delta\eta^\alpha$ constant increments of the coordinates and $\eta_o^{\,\alpha}$ the coordinates of a conveniently chosen origin.

The dependence on ζ is made discrete by conceptually dividing the shell into K layer of equal thickness. Within each layer those variables defined outside the middle surface are assumed constant with respect to ζ. Numbering each layer in sequence, $k = 1, \cdots K$, the variables become functions of the layer number k, as well as functions of (m,n). When needed, the value of ζ in each layer is taken to be its mid-layer value:

$$\zeta = (k - \frac{K+1}{2})\,\frac{h}{K} \; ; \quad k = 1, \cdots K, \qquad (2.2)$$

where h is the thickness of the shell.

The principal reason for making the dependence on ζ discrete is in order to model the variations of the stresses through the thickness. For, while the functional form of the dependence of the strains on ζ is prescribed by the Kirchhoff hypothesis *, no such simplification is possible for the stresses. Plasticity makes the *a priori* determination of the dependence of the stresses on ζ not feasible.

Strain hardening behavior is modelled using the "mechanical sublayer model" **: the stress tensor at a point is assummed to be a weighted sum of J "sublayer" stress tensors or, more simply, J "substress" tensors, with each sublayer obeying the same linear incremental stress-strain relation, but having different yield stresses. Using this model, there will be K stress tensor values at each mesh point (m,n) and J×K substress tensor values.

* Cf equation (2.15) of this report.

** This method of modelling strain hardening behavior has been developed and extensively used at the Aeroelastic and Structure Research Lab. of MIT. A detailed description of the method is given in [2; Sect. 5.4.2].

The dependence on t, the time, is made discrete by replacing time derivatives by equivalent finite difference operators. As already mentioned, an explicit finite difference scheme results. The scheme characterizes the change undergone by the dependent variables during a time interval Δt. A constant time interval is used, which is automatically determined by the program to assure numerical stability.

The basic function of the finite difference scheme is to advance the values of the following fundamental variables.

- $y^i(m,n)$*, the Cartesian coordinates of the middle surface at mesh point (m,n);
- $n^i(m,n)$, the Cartesian components of the unit normal to the middle surface at mesh point (m,n);
- $\Delta u^i(m,n)$, the Cartesian components of the displacement increment undergone by the middle surface at mesh point (m,n) during the time interval Δt;
- $\sigma^{\alpha\beta}(m,n,k)$, the contravariant components of the stress tensor (or, in the case of strain hardening, the substress tensor) tangential to the middle surface at mesh point (m,n) at layer (or sublayer) station k.

The dependence of the variables on (m,n) or (m,n,k) is indicated explicitly to emphasize that their values are stored as 2- and 3-dimensional arrays in the program. From these fundamental variables all the other dependent variables of the theory are determined.

For the sake of clarity, the description of the computational algorithm used by REPSIL is separated into four groups of calculations:

- Initialization calculations,
- Finite difference calculations,
- Energy calculations,
- Surface strains calculations.

Within each group the description is organized by units of calculations. Units can occur within the main program or as subroutines. In the latter case, the subroutine description, which follows the subroutine name, is indented. When a subroutine occurs within a subroutine, the description

---

*Latin indices indicate Cartesian components in Euclidean 3-space and range over the integers 1, 2, and 3.

of the embedded subroutine is further indented. All subroutine names are completely capitalized.

2.2 Initialization Calculations

The initialization calculations determine the program constants and the initial values of the dependent variables from the input data. The flow chart in Figure 2.1 summarizes the calculations, whose description follows.

START Figure 2.2 outlines this subroutine. The input data, including material properties, is read into storage and program constants are calculated. The minimum and maximum values of the mesh numbers are determined from the boundary conditions. Subroutine INGEOM is called within this subroutine.

INGEOM This subroutine generates the initial geometry of the shell. First, the dimensions characterizing the particular geometry being treated are read off the input cards. Next, the finite difference increments $\Delta\eta^{\alpha}$ are calculated. Finally, the subroutine computs the initial Cartesian coordinates $y^{i}(m,n)$ of the middle surface and stores the resulting arrays. The user not finding a suitable geometry among those described in Section 3.2 (flat plate, cylinder or cone) is referred to Section 6.4 for instructions on programming initial geometries.

Returning to START, the program calculates the interpolation coeffienents at locations where STRAIN * computes the surface strains and the mesh number of the mesh point bracketing these locations. Next, a stable time interval Δt is found from the criteria

$$\Delta t_M = \frac{2}{c}\left[\frac{1}{(\Delta\eta^1)^2} + \frac{1}{(\Delta\eta^2)^2}\right]^{-\frac{1}{2}}, \quad c = \sqrt{\frac{E}{\rho(1-\nu^2)}}\,; \tag{2.3}$$

$$\Delta t_B = \frac{1}{2hc}\left[\frac{1}{12}\left(1-\frac{1}{K^2}\right)\left(\frac{1}{(\Delta\eta^1)^4} + \frac{1}{(\Delta\eta^2)^4} + \frac{1+15\nu}{8(\Delta\eta^1\Delta\eta^2)^2}\right)\right]^{-\frac{1}{2}}.$$

These criteria result from a von Neumann stability analysis of an elastic flat plate **; Δt_M is the maximum stable interval for membrane motion and Δt_B for bending motion. The program

---

* Subroutine STRAIN is described in Section 2.5.

** A report describing this analysis is being prepared.

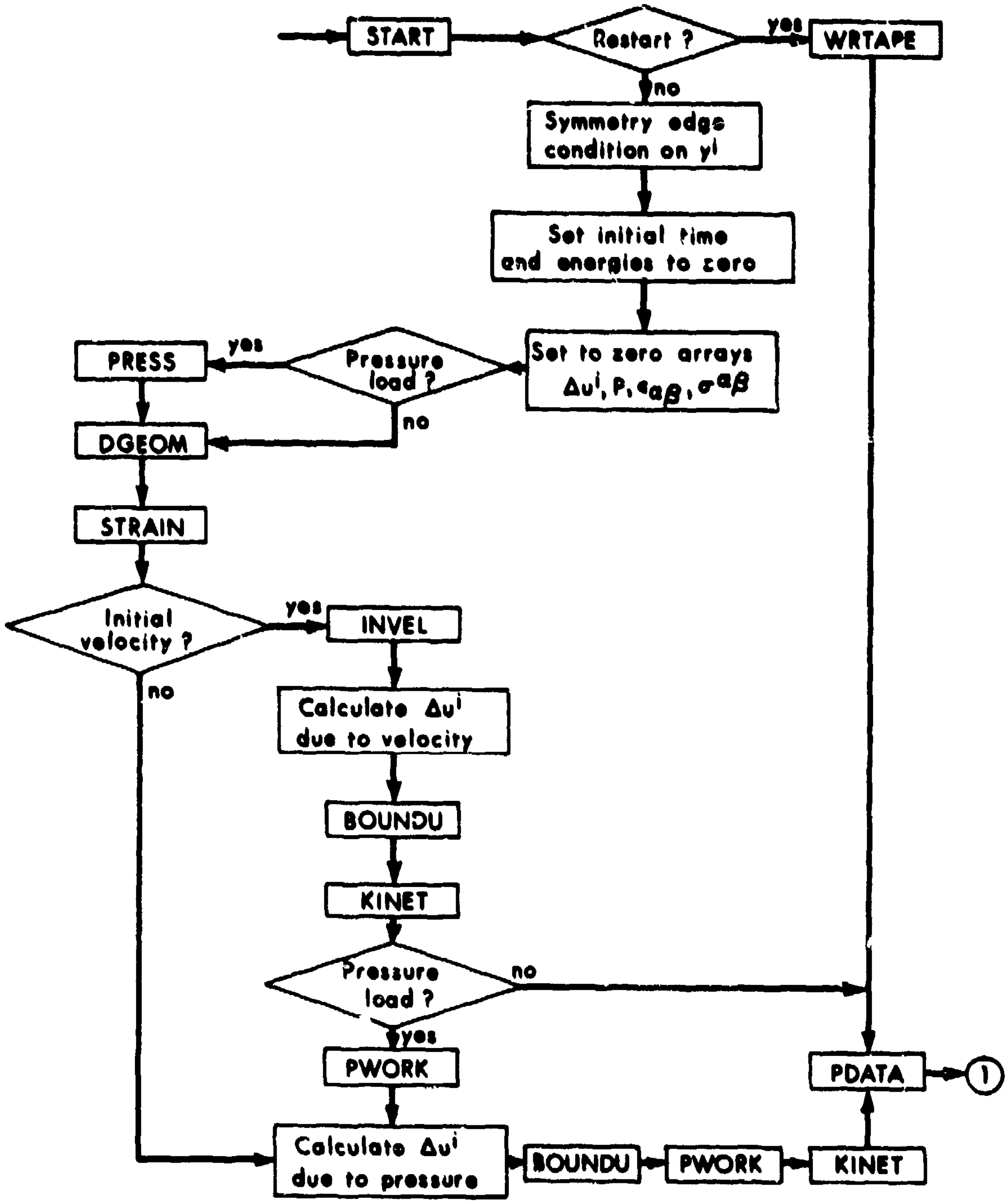


Figure 2.1 Flow Chart for Initialization Calculations

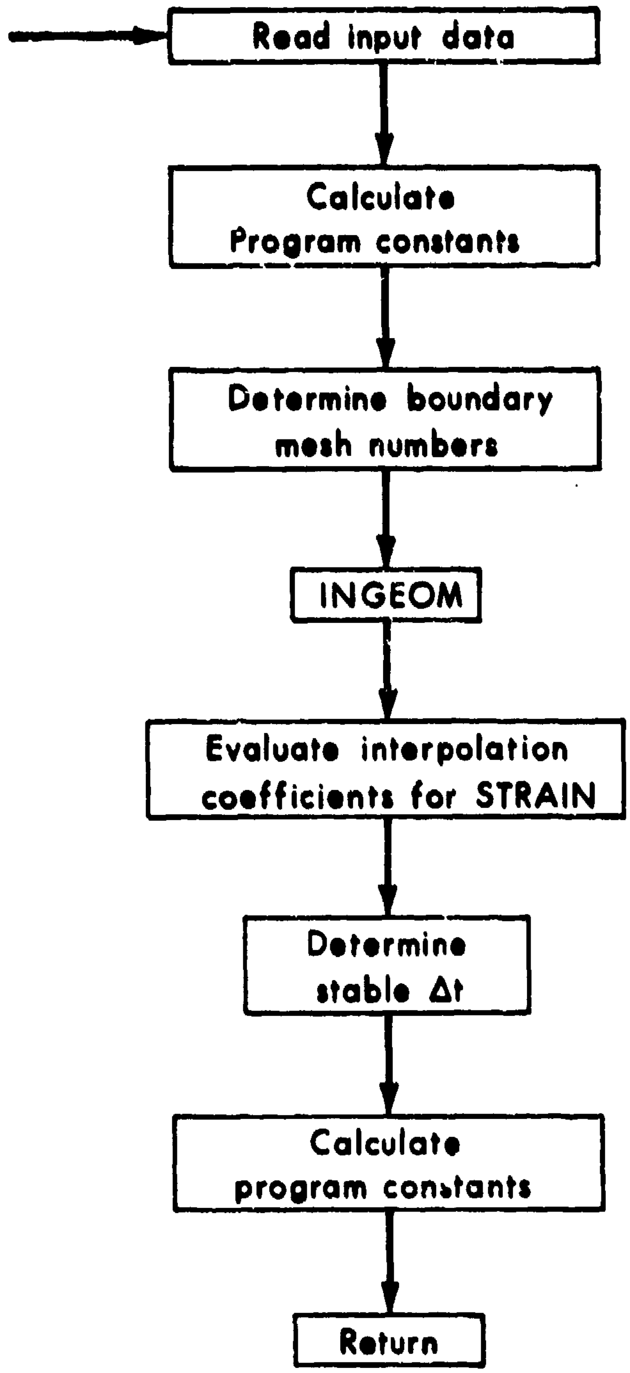


Figure 2.2 Flow Chart for Subroutine START

selects for use in the finite difference calculations the minimum of Δt_M, Δt_B and the Δt prescribed in the input data.

The subroutine ends with the calculation of the remaining program constants, most of which require for their determination the values of $\Delta\eta^{\alpha}$ computed in INGEOM.

The main program next determines whether the problem being solved is a new one, starting from an initial (stress-free) configuration at time $t_0 = 0$ or a continuation of a problem whose solution up to some time $t_\ell > 0$ has already been calculated, called a restart problem or, simply, a restart. If it is a restart, subroutine WRTAPE is called in order to read data off the restart tape, after which subroutine PDATA is called and the initialization calculations end (see Figure 2.1).

WRTAPE Depending on the instructions of the main program, this subroutine either reads off or writes on a tape, the restart tape, the values of the fundamental variables y^i, n^i, Δu^i and $\sigma^{\alpha\beta}$, the surface strains and the energy variables at time step prescribed by the initial data. This information is sufficient to permit the program to continue the solution of the problem from any of the prescribed time steps. During the initialization portion of the program this subroutine reads this information off the tape and during the finite difference portion it stores the information.

PDATA This subroutine stores on a tape, the plotting tape, the data required by the REPSIL plotting program. A description of these data are given in Section 4.2. Appendix D describes and lists the plotting program.

If, on the other hand, the program is tr ting a new problem, WRTAPE is not called at all and PDATA is not call intil later (see Figure 2.1). Rather, for a new problem the program } eeds to generate the information otherwise obtained from WRTAPE.

As discussed in Section 3.2, the program treats symmetric deformations about one or two planes. For such deformations, the shell variables, in particular the middle surface coordinates y^i, are symmetrically distributed about symmetry planes. The program imposes this condition numerically by relating the coordinates one mesh spacing outside the symmetry edge to the coordinates one mesh spacing inside the edge. Typically, for the symmetry plane located in the y^2, y^3 coordinate plane and intersecting the middle surface along the curve with mesh number m (see Figures 3.6 - 3.8 where m = 2) these relations are

$$y^1(m-1,n) = -y^1(m+1,n)$$

$$y^2(m-1,n) = y^2(m+1,n)$$

$$y^3(m-1,n) = y^3(m+1,n)$$

$$y^1(m,n) = 0, \tag{2.4}$$

for all allowable n.

The initial time t_0, kinetic energy T, strain energy V, damping work W_D and external work W are next set equal to zero. This is followed by the arrays for the displacement increments Δu^i, the pressure P, the tangential strain components $\varepsilon_{\alpha\beta}$ and the tangential stress components all being cleared to zero. If pressure loads are acting on the shell, the initial pressure distribution is obtained from subroutine PRESS, whose description is postponed until Section 2.3. If no pressure loads are acting, the program skips PRESS and goes directly to DGEOM. DGEOM is described in Section 2.3, where it is shown that at this point in the program it is called in order to calculate the normal n^i, the augmented pressure P*, the time constant $\Delta t^2/(a_0^{\frac{1}{2}} \Gamma_0)$ and the initial values of the tensor $a_{\alpha\beta}$ and $b_{\alpha\beta}$. Next the program goes to STRAIN where the values of $a_{\alpha\beta}$ and $b_{\alpha\beta}$ are interpolated to give the metric tensor $G_{\alpha\beta}$ on the bounding surfaces of the shell; the details of this calculation are given in Section 2.5.

The remainder of the initialization calculations are concerned with obtaining the initial values of the displacement increment arrays $\Delta u^i(m,n)$ and the associated values of the kinetic energy T and external work W. The initial displacement increments can arise in three ways: (1) from an initial impulse velocity distribution; (2) from an initial pressure distribution or (3) from a combination of both. Should they be due to an initial velocity distribution or combination, the program calls INVEL.

INVEL This subroutine determines the magnitude of the initial velocity v at each mesh point and multiplies this by the normal $n^i(m,n)$ to give the initial velocity distribution:

$$v^i(m,n) = v\, n^i(m,n). \tag{2.5}$$

The initial velocity magnitudes can either be read off input data cards, as shown in Section 3.2, or the user can program an analytical expression for these magnitudes, as shown in Section 6.2.

The program then calculates the displacement increments due to the initial velocity distribution

$$\Delta u^i(m,n) = v^i(m,n)\,\Delta t \qquad (2.6)$$

and proceeds to subroutine BOUNDU. This subroutine, described in Section 2.3, adjusts the displacement increments one mesh space in from clamped edge boundaries so that clamped edge conditions are satisfied and also generates the displacement increment one mesh spacing outside symmetry boundaries from the values one mesh spacing inside, using equations for Δu^i very much like (2.3) for y^i. Next the kinetic energy T associated with the initial displacement increment distribution is determined in subroutine KINET, described in Section 2.4. If there is no initial pressure distribution, the displacement increments $\Delta u^i(m,n)$ and kinetic energy just determined are associated with the time interval from $t_o = 0$ to $t_1 = \Delta t$ and the initialization calculations terminate with the calling of subroutine PDATA. On the other hand, if the shell is also subjected to an initial pressure distribution, the displacement increments $\Delta u^i(m,n)$ and kinetic energy T just determined are associated with the time interval from $t_{-1} = -\Delta t$ to $t_o = 0$ * and the external work due to the pressure distribution acting on the displacement increments just determined is computed in subroutine PWORK, described in Section 2.4. The program then proceeds to calculate the displacement increments for the time interval $[t_o, t_1]$ using an equation which is nothing more than the equation of motion (2.35) with the stresses set equal to zero:

$$\Delta u^i_+ = \Delta u^i - \frac{\Delta t^2}{r_o^{\frac{1}{2}}\,\Gamma_o}\,P^*\,n^i\,, \qquad (2.7)$$

with Δu^i_+ and Δu^i the displacement increments for the time intervals $[t_o, t_1]$ and $[t_{-1}, t_o]$, respectively. When there is no initial impulse velocity distribution, but only a pressure distribution the program goes from STRAIN to directly computing the displacement increments for the interval $[t_o, t_1]$ using (2.7), with, however, the displacement

---

* The reason that the displacement increments were previously associated with the time interval $[t_o, t_1]$ rather than the interval $[t_{-1}, t_o]$ is that with no pressure acting and the shell being stress free at the time t_o, the displacement increments are the same for both intervals. On the other hand, a pressure distribution at time t_o will cause the displacement increments for $[t_o, t_1]$ to change from those for $[t_{-1}, t_o]$.

increments for $[t_{-1}, t_o]$ set equal to zero.

Again, the new Δu^i are adjusted in BOUNDU to satisfy clamped edge and symmetry edge conditions and the additional external work resulting for the initial pressure distribution acting on the displacement increments for the interval $[t_o, t_1]$ is computed in PWORK. Lastly, the kinetic energy during the time interval $[t_o, t_1]$ is calculated in KINET and the program proceed to PDATA where information is gathered for the plotting program. With this last operation the initialization calculation end and the program proceeds to the finite difference calculational loop.

2.3 Finite Difference Calculations

The finite difference calculations follow the initialization calculations and are repeated each time step. They solve the finite difference equations for the current values of the fundamental variables. In this way, the values of these variables are advanced each time step and the history of the deformation is generated.

Aside from minor simplifications in notation, the finite difference equations presented in this section are identical with those given in [1; Sect. 7.3]. They are written in compact form with only finite time derivatives being shown explicitly; finite differences with respect to material coordinates η^α are symbolically represented by their corresponding partials. At interior points and along symmetry boundaries the program uses central difference operators, while along hinged and clamped boundaries it uses forward or backward difference operator, all operators being of order $|\Delta\eta^\alpha|^2$ accuracy *.

Figure 2.3 outlines the finite difference calculations. They comprise a sequence or, better still, a loop of calculations, which, as already mentioned, are repeated every time step. The description of this loop begins after ℓ-1 time steps have elapsed, at the generic time $t_- = (\ell-1)\ \Delta t$. The values of the middle surface coordinates, the normal to the middle surface and the tangential stress or substress component are assumed known at this time and are denoted by appending subscripted minus signs: y^i_-, n^i_- and $\sigma^{\alpha\beta}_-$. Also the components of the displacement increment for the next time interval, from the time $t_- = (\ell-1)\ \Delta t$ to the time $t = \ell\Delta t$, are known and are denoted simply as Δu^i. Using these values of the fundamental variables, the finite difference calculations generate suceeding values of the fundamental

---

* Appendix A summarizes the explicit forms of these operators.

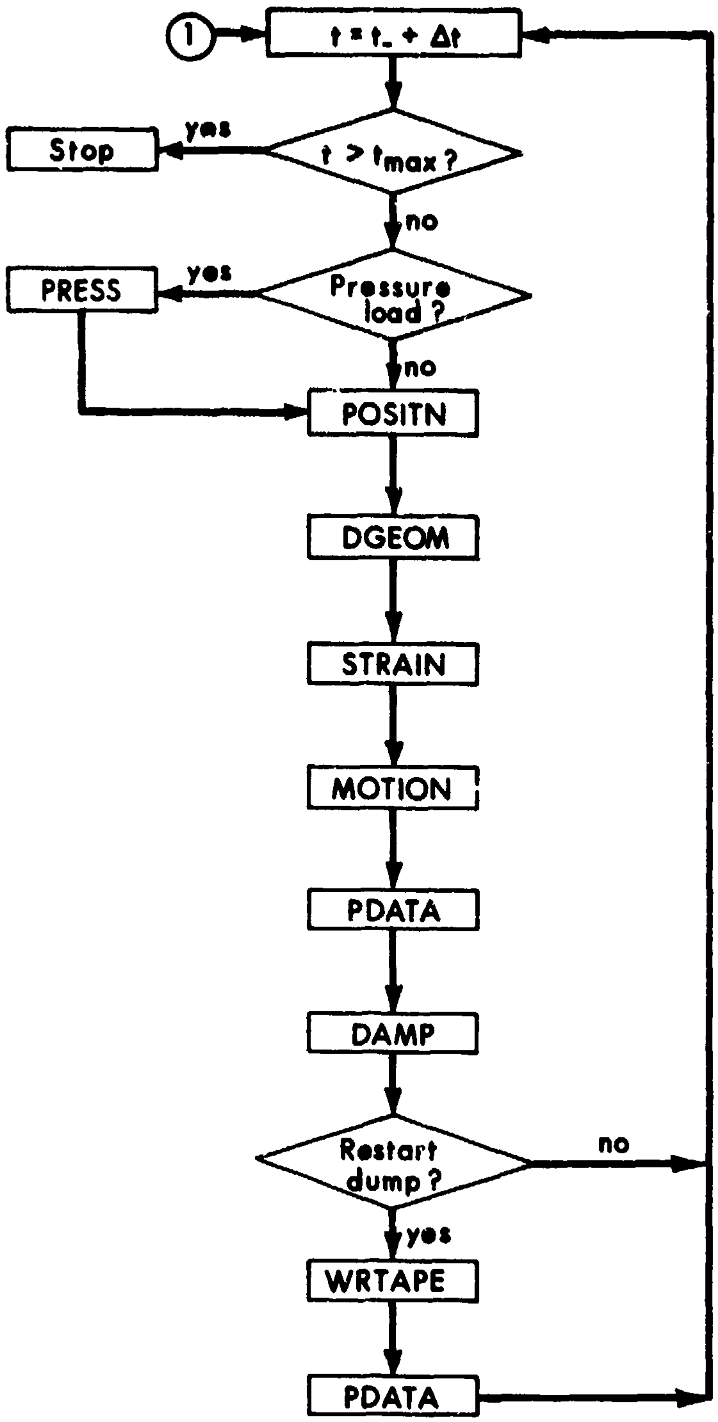


Figure 2.3 Flow Chart for Finite Difference, Energy and Surface Strain Calculations

variable y^i, n^i, $\sigma^{\alpha\beta}$ and Δu_+^i at the time $t = \ell\Delta t$ as follows.

First the time is updated:

$$t = t_- + \Delta t \ , \tag{2.8}$$

and checked against a prescribed maximum, t_{max}. If the maximum is exceeded, then the solution has progresses to completion and calculations stop. Otherwise, the program checks as to whether there is a pressure loading acting on the shell at this time; if there is then the program calls subroutines PRESS, POSITN, DGEOM, STRAIN, MOTION, PDATA and DAMP in that order; if not, it skips PRESS and calls the remainder of the sequence.

PRESS This subroutine supplies the values of pressure at mesh intersection points for the given time and stores them in the array P(m,n). The user is expected to supply this information either by programming some analytical expression for the pressure, as outlined in Section 6.3, or by generating a tape with this data in numerical form.

POSITN This subroutine simply calculates the coordinates of the middle surface at the current time t using the formula

$$y^i = y_-^i + \Delta u^i \ , \tag{2.9}$$

storing the values in the arrays $y^i(m,n)$.

DGEOM This subroutine is summarized in Figure 2.4. It begins by calling subroutine GRAD.

GRAD This subroutine, using appropriate finite difference operators, determines the first and second gradients of y^i and Δu^i with respect to the material coordinates η^α:

$$\begin{aligned} y_\alpha^i &= \frac{\partial y^i}{\partial \eta^\alpha} \ , & y_{\alpha\beta}^i &= \frac{\partial^2 y^i}{\partial \eta^\alpha \partial \eta^\beta} \ , \\ u_\alpha^i &= \frac{\partial \Delta u^i}{\partial \eta^\alpha} \ , & u_{\alpha\beta}^i &= \frac{\partial^2 \Delta u^i}{\partial \eta^\alpha \partial \eta^\beta} . \end{aligned} \tag{2.10}$$

After GRAD the program stores the preceding values of the components of the normal n_-^i for later use in calculating strain increments. From the first and second gradient of y^i, the program determines the differential geometric quantities characterizing

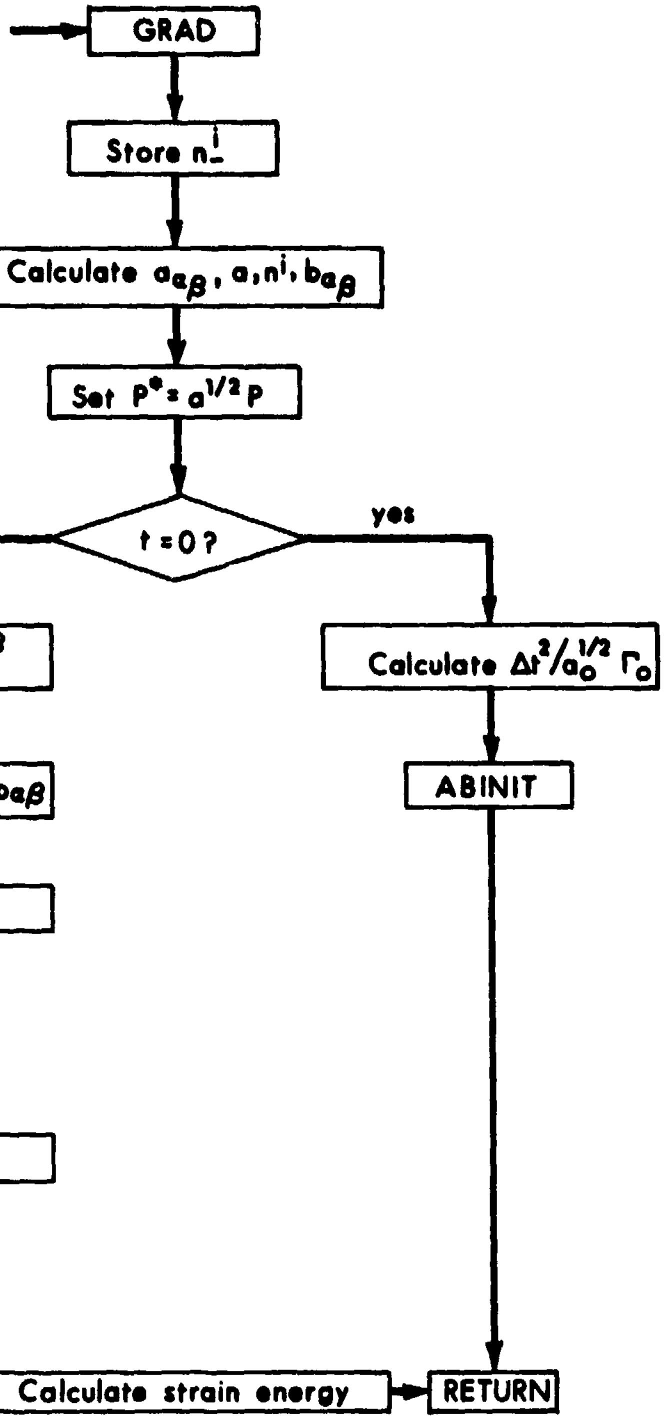


Figure 2.4 Flow Chart for Subroutine DGEOM

the current position of the middle surface: the covariant components of the metric $a_{\alpha\beta}$, the second fundamental tensor $b_{\alpha\beta}$ and the current normal n^i, and also the determinant a of the metric as follows.

$$a_{\alpha\beta} = y^i_\alpha \, y^{i*}_\beta \quad , \quad a = a_{11} \, a_{22} - (a_{12})^2 \quad ,$$
$$n^i = e^{ijk} y^j_1 \, y^k_2 \, / \, a^{\frac{1}{2}} \quad , \quad b_{\alpha\beta} = n^i \, y^i_{\alpha\beta} \quad , \qquad (2.11)$$

where e^{ijk} is the permutation symbol.** Next, the pressure distribution is modified for use in the equation of motion:

$$P^* = a^{\frac{1}{2}} \, P. \qquad (2.12)$$

The remainder of the subroutine's calculations depend on the time. Since initially the shell is assumed to be in a stress-free state, there is no need to calculate the stress field in DGEOM. Rather, the subroutine determines certain time constants and stores these for later use: $\Delta t^2/(a_o^{\frac{1}{2}} \, \Gamma_o)$ is calculated for use in the equations of motion and the expression for the kinetic energy and the initial values of $a_{\alpha\beta}$ and $b_{\alpha\beta}$ at prescribed mesh points are selected using subroutine ABINIT for use in STRAIN, as described in Section 2.5. At all subsequent times, the subroutine calculates the stress field and other stress related quantities as follows. First, the contravariant components $a^{\alpha\beta}$ of the metric, the Christoffel symbols $\Gamma_{\alpha\beta}{}^\gamma$ and the mixed components b^α_β of the second fundamental tensor are calculated:

$$a^{11} = a_{22}/a, \; a^{12} = -a_{12}/a, \; a^{22} = a_{11}/a,$$
$$\Gamma_{\alpha\beta}{}^\gamma = a^{\gamma\delta} \, y^i_\delta \, y^i_{\alpha\beta} \, , \quad b^\alpha_\beta = a^{\alpha\delta} \, b_{\delta\beta} \qquad (2.13)$$

* We introduce the summation convention: terms or products of terms having the same index appearing twice are to be summed over the range of the index. In the case of repeated Latin indices both will be superscript since their basis is Cartesian, while repeated Greek indices will always appear as paired superscript and subscript.

** That is, $e^{ijk} = 1$ for i, j, k an even permutation of 1,2,3; = -1 for i, j, k an odd permutation; = 0 for i, j, k non-distinct.

Next the incremental changes in $a_{\alpha\beta}$ and $b_{\alpha\beta}$ due to the incremental displacement Δu^i are determined:

$$\Delta n_\alpha = -n_-^i \, u_\alpha^i \;, \quad \Delta n = \frac{a^{\alpha\beta}\Delta n_\alpha \Delta n_\beta}{1 + n^i \, n_-^i} \;,$$

$$\Delta a_{\alpha\beta} = y_\alpha^i \, u_\beta^i + y_\beta^i \, u_\alpha^i - u_\alpha^i \, u_\beta^i \,. \qquad (2.14)$$

$$\Delta b_{\alpha\beta} = n_-^i \, u^i{}_{\alpha\beta} + \Gamma_{\alpha\beta}{}^\gamma \, \Delta n_\gamma + b_{\alpha\beta}\Delta n.$$

These expressions are exact with no approximations based on the smallness of Δu^i being used. The corresponding increments undergone by the tangential strain components $\varepsilon_{\alpha\beta}$ at the station $\zeta(k)$ distant from the middle surface are then calculated using the equation

$$\Delta\varepsilon_{\alpha\beta} = \tfrac{1}{2}\Delta a_{\alpha\beta} - \zeta\Delta b_{\alpha\beta} \qquad (2.15)$$

This equation is based on the thin shell approximation *. Subroutine STRESS follows.

STRESS The principal function of this subroutine, outlined in Figure 2.5, is to calculate the current stress or substress component from their preceding values and the incremental change in the strain component. Calculations begin by evaluating the metric $g_{\alpha\beta}$ and inverse $g^{\alpha\beta}$ for the lamella ζ distance from the middle surface:

$$g_{\alpha\beta} = a_{\alpha\beta} - 2\zeta \, b_{\alpha\beta} \;, \quad g = g_{11} \, g_{22} - (g_{12})^2 \,,$$

$$g^{11} = g_{22}/g \;, \; g^{12} = -g_{12}/g \;, \; g^{22} = g_{11}/g \,. \qquad (2.16)$$

With these terms evaluated, the mixed components of the incremental strain and preceding stress are immediately obtained:

$$\Delta\varepsilon^\alpha_\beta = g^{\alpha\delta}\Delta\varepsilon_{\delta\beta} \;, \quad \sigma_-{}^\alpha{}_\beta = g_{\beta\delta}\sigma_-^{\delta\alpha} \qquad (2.17)$$

Next, assuming that the incremental deformation is elastic, the increments in the stress components are calculated using the linear isotropic law:

---

* Cf [1; Sect. 7.1].

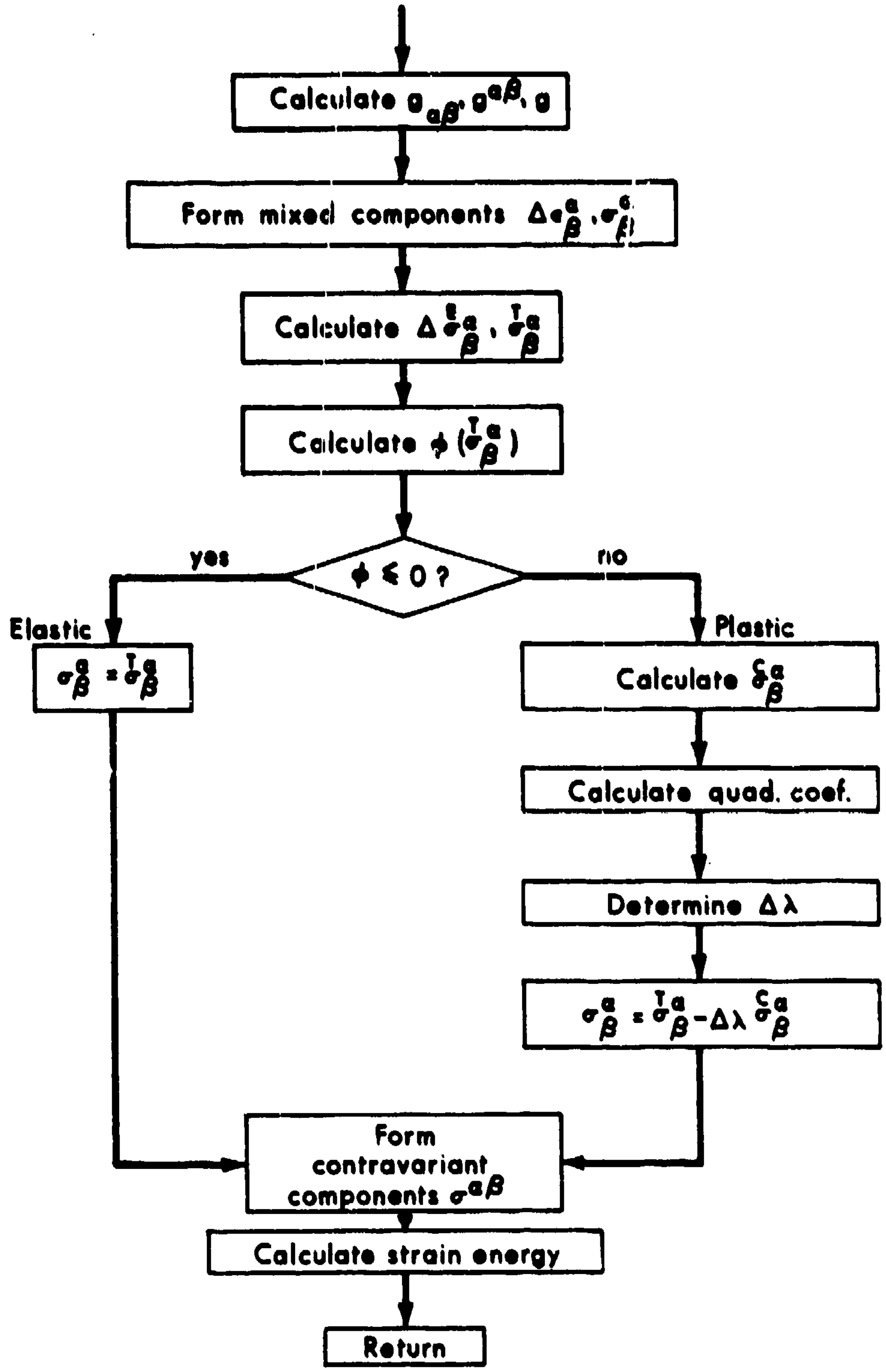


Figure 2.5 Flow Chart for Subroutine STRESS

$$\Delta\overset{E\alpha}{\sigma}_{\beta} = \frac{E}{1+\nu} \left(\Delta\varepsilon^{\alpha}_{\beta} + \frac{\nu}{1+\nu} \; \delta^{\alpha}_{\beta} \; \Delta\varepsilon^{\gamma}_{\gamma}\right) . \qquad (2.18)$$

In the case of strain hardening, $\Delta\overset{E\alpha}{\sigma}_{\beta}$ are elastic increments in the substress components and are the same for each layer. These increments are added to the preceding values to give the components of a trial stress or substress

$$\overset{T\alpha}{\sigma}_{\beta} = \sigma^{\alpha}_{-\beta} + \Delta\overset{E\alpha}{\sigma}_{\beta} . \qquad (2.19)$$

The trial stress is tested against the yield function

$$\Phi(\overset{T\alpha}{\sigma}_{\beta}) \equiv \frac{3}{2} \; \overset{T\alpha}{\sigma}_{\beta}\overset{T\beta}{\sigma}_{\alpha} - \frac{1}{2} \left(\overset{T\alpha}{\sigma}_{\alpha}\right)^{2} - \sigma_{o}^{2} \quad , \qquad (2.20)$$

where σ_o is the uniaxial yield stress or, in the case of strain hardening, the yield substress; for rate sensitive behavior σ_o is assumed to depend on the second invariant of the strain rate deviator:

$$\sigma_o = \sigma_{o(\text{static})}\left[1 + \left(\frac{\dot{\varepsilon}}{d}\right)^{\frac{1}{p}}\right] , \qquad (2.21)$$

with

$$\dot{\varepsilon} = \frac{1}{\Delta t} \left[\frac{3}{2} \; \Delta\varepsilon^{\alpha}_{\beta}\Delta\varepsilon^{\beta}_{\alpha} - \frac{1}{2} \left(\Delta\varepsilon^{\gamma}_{\gamma}\right)^{2}\right]^{\frac{1}{2}} . \qquad (2.22)$$

If $\Phi(\overset{T\alpha}{\sigma}_{\beta}) \leq o$, then the trial stress is on or within the yield surface and its components define acceptable values for the current stress components $\sigma^{\alpha}_{\beta} = \overset{T\alpha}{\sigma}_{\beta}$. Hence, the stress increment is indeed elastic and the calculations for a plastic stress increment are skipped, the program proceeding directly to (2.29) below. On the other hand, if $\Phi(\overset{T\alpha}{\sigma}_{\beta}) > o$, then the trial stress is outside the yield surface and, hence, unacceptable. In this case, the subroutine determines a correction to the trial stress due to plastic flow. First, the components of a corrector stress are determined:

$$\overset{D\alpha}{\sigma}_{\beta} = 3 \; (1-\nu)\sigma^{\alpha}_{-\beta} - (1-2\nu)\sigma^{\gamma}_{-\gamma} \; \delta^{\alpha}_{\beta} \qquad (2.23)$$

The corrector stress gives the direction in stress space in which to apply a correction to ${}^{T\alpha}\sigma_{\beta}$ in order to bring the resultant stress components

$$\sigma^{\alpha}_{\beta} = {}^{T\alpha}\sigma_{\beta} - \Delta\lambda\ {}^{D\alpha}\sigma_{\beta} \tag{2.24}$$

back to the yield surface, so that

$$\Phi({}^{T\alpha}\sigma_{\beta} - \Delta\lambda\ {}^{D\alpha}\sigma_{\beta}) = 0. \tag{2.25}$$

This gives a quadratic equation in $\Delta\lambda$

$$A\ \Delta\lambda^2 - 2\ B\ \Delta\lambda + C = 0\ , \tag{2.26}$$

where

$$\begin{aligned} A &= {}^{D\alpha}\sigma_{\beta}\ {}^{D\beta}\sigma_{\alpha} - \frac{1}{3}\ {}^{D\alpha}\sigma_{\alpha}\ {}^{D\beta}\sigma_{\beta} \\ B &= {}^{D\alpha}\sigma_{\beta}\ {}^{T\beta}\sigma_{\alpha} - \frac{1}{3}\ {}^{D\alpha}\sigma_{\alpha}\ {}^{T\beta}\sigma_{\beta} \\ C &= {}^{T\alpha}\sigma_{\beta}\ {}^{T\beta}\sigma_{\alpha} - \frac{1}{3}\ {}^{T\alpha}\sigma_{\alpha}\ {}^{T\beta}\sigma_{\beta} - \frac{2}{3}\sigma_o{}^2\ , \end{aligned} \tag{2.27}$$

which the program solves for the smallest positive value of $\Delta\lambda$:

$$\Delta\lambda = B - (B^2 - AC)^{1/2}\ .^{*} \tag{2.28}$$

With $\Delta\lambda$ known, current stress or substress components are determined from (2.24) and are then put into contravariant form

$$\sigma^{\alpha\beta} = g^{\alpha\delta}\ \sigma^{\beta}_{\delta}\ . \tag{2.29}$$

---

* If $\Delta\lambda$ turns out to be negative or complex, then the subroutine uses a procedure described in [3; Sect. IV], which divides the elastic increment components $\Delta{}^{E\alpha}\sigma_{\beta}$ into L elastic subincrements and applys a correction to each step such that $\Delta\lambda$ is always positive.

In the case of strain hardening, a weighted sum of the substress components of the layer is taken to obtain the layer stress components before being put in contravariant form. The subroutine finishes by computing the contribution to the strain energy of the current stress using (2.39) of Section 2.4 and returning to DGEOM.

Next, the components of strain at prescribed locations on the bounding surfaces of the shell are evaluated using (2.55) as explained in Section 2.5. Subroutines RESULT and SYMTRY are called next.

RESULT This subroutine numerically integrates the stress components (note: not the substress components) and their moments through the thickness to give components of the membrane and bending resultants:

$$Q^{\alpha\beta} = a^{\frac{1}{2}} \sum_k \sigma^{\alpha\beta} (1-\zeta\, b^{\gamma}_{\gamma})\, \Delta\zeta \quad ,$$

$$M^{\alpha\beta} = a^{\frac{1}{2}} \sum_k [\sigma^{\alpha\beta}(1-\zeta\, b^{\gamma}_{\gamma}) - \frac{\zeta}{2} (\sigma^{\alpha\delta}\, b^{\beta}_{\delta} + \sigma^{\beta\delta}\, b^{\alpha}_{\delta})]\zeta\, \Delta\zeta \quad . \qquad (2.30)$$

From these components the subroutine determines the components of the stress resultant:

$$N^{i\alpha} = Q^{\alpha\beta}\, y^{i}_{\beta} + \Gamma_{\beta\gamma}{}^{\alpha} M^{\beta\gamma}\, n^{i} \quad . \qquad (2.31)$$

SYMTRY This subroutine imposes the symmetry edge conditions on n^i, $M^{\alpha\beta}$ and $N^{i\alpha}$ relating the values of these variables one mesh spacing outside the symmetry edge to their values one mesh spacing inside. Typically, for the symmetry plane located in the y^2, y^3 coordinate plane and intersecting the middle surface along the m equals a constant curve these relations are:

$$n^1(m-1,n) = -n^1(m+1,n) \quad ,$$

$$n^2(m-1,n) = n^2(m+1,n) \quad , \qquad (2.32)$$

$$n^3(m-1,n) = n^3(m+1,n) \quad ,$$

$$M^{11}(m-1,n) = M^{11}(m+1,n) \quad ,$$

$$M^{12}(m-1,n) = -M^{12}(m+1,n) \quad , \qquad (2.33)$$

$$M^{22}(m-1,n) = M^{22}(m+1,n) \quad ,$$

$$N^{11}(m-1,n) = N^{11}(m+1,n) \quad ,$$

$$N^{21}(m-1,n) = -N^{21}(m+1,n) \quad ,$$

$$N^{31}(m-1,n) = -N^{31}(m+1,n) \quad ,$$

$$(2.34)$$

$$N^{12}(m-1,n) = -N^{12}(m+1,n) \quad ,$$

$$N^{22}(m-1,n) = N^{22}(m+1,n) \quad ,$$

$$N^{32}(m-1,n) = N^{32}(m+1,n) \quad ,$$

for all admissible n. These relations as well as relations for other symmetry plane locations used in the program are derived in [1; Sect. 6.4 and App. B].

Returning to DGEOM, the program determines the strain energy by summing the contribution of each mesh point and layer as described in Section 2.5. This ends DGEOM and calculations return to the main program.

Next, the strain components at selected points on the bounding surfaces are determined in subroutine STRAIN, as described in Section 2.5, after which subroutine MOTION is called.

MOTION This subroutine is summarized in Figure 2.6. Its principal function is to determine the values Δu_+^i of the components of the displacement increments undergone by the middle surface in the time interval $[t, t + \Delta t]$. First a check is performed to determine whether a pressure distribution is currently acting on the shell. If a pressure distribution is acting, then subroutine PWORK, which is described in Section 2.4, is called in order to determine the contribution to the external work of the pressure acting through the displacement increments Δu^i of the time interval $[t - \Delta t, t]$. With no pressure loads acting, the subroutine skips PWORK and proceeds directly to determine Δu_+^i from the finite difference form of the equations of motion

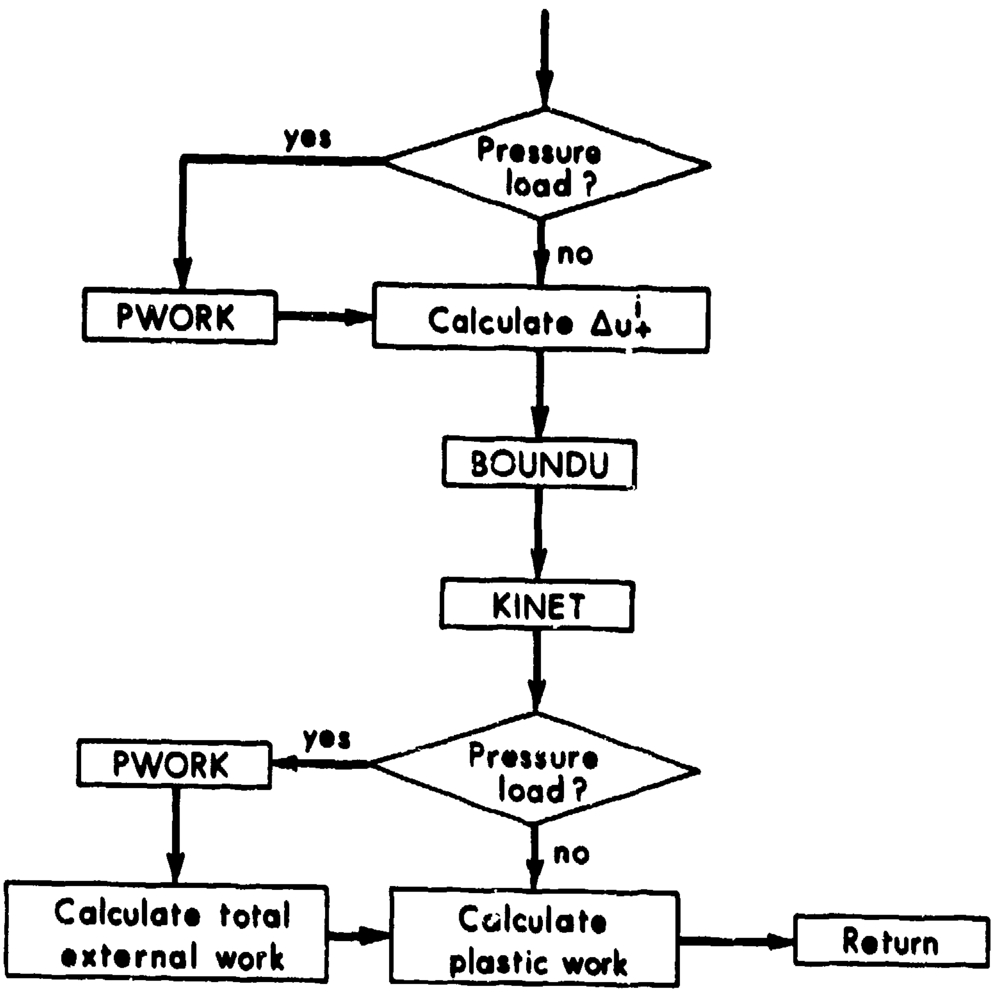


Figure 2.6 Flow Chart for Subroutine MOTION

$$a_o^{1/2}\left[\Gamma_o \frac{\Delta u_+^i - \Delta u^i}{\Delta t^2} + D\,\frac{\Delta u_+^i + \Delta u^i}{2\,\Delta t}\right]$$

$$= \frac{\partial^2 (M^{\alpha\beta} n^i)}{\partial\eta^\alpha \partial\eta^\beta} + \frac{\partial N^{i\alpha}}{\partial\eta^\alpha} - P^* n^i . \qquad (2.35)$$

As indicated earlier, the partials appearing in the right hand side of these equations represent finite difference derivatives. These equations do not coincide with those given in [1; Eq. (7.38)] due to the addition of the term $D\,\frac{\Delta u_+^i + \Delta u^i}{2\,\Delta t}$, representing a linear viscous damping effect. However, this damping term only appears when the damping option is used otherwise, the term is set equal to zero. Next, subroutine BOUNDU is called.

BOUNDU This subroutine generates the additional values of Δu_+^i need along symmetry edges and modifies the values of Δu_+^i along clamped edges so that clamped edge conditions are satisified. As already described in subroutine SYMTRY, the values of variables one mesh spacing outside a symmetry edge are related to their values one mesh spacing in. The conditions imposed on Δu_+^i are similar to those imposed on y^i in (2.4): for the symmetry plane located in the y^2, y^3 coordinate plane intersecting the middle surface along the m equals a constant curve

$$\begin{aligned} \Delta u_+^1\,(m-1,n) &= -\Delta u_+^1\,(m+1,n) , \\ \Delta u_+^2\,(m-1,n) &= \Delta u_+^2\,(m+1,n) , \\ \Delta u_+^3\,(m-1,n) &= \Delta u_+^3\,(m+1,n) , \\ \Delta u_+^1(m,n) &= 0 \end{aligned} \qquad (2.36)$$

for all admissible n. Along clamped edges the components of the normal n^i must remained fixed at their initial values. This condition is achieved by adjusting Δu_+^i one mesh spacing in from the clamped edge so that its component in the direction of the edge normal is 1/4 the value of the corresponding component of Δu_+^i two mesh spacings in from the edge. This adjustment guarantees

that the tangent to the middle surface at a clamped edge, which is computed using a one-sided finite derivative, will always lie in a fixed plane perpendicular to the original normal. Typically, for a clamped edge fixed in the y^3, y^1 coordinate plane intersecting the middle surface along a curve with fixed mesh number n, the relations used are

$$\Delta u_+^i \ (m,n+1) = \Delta u_*^i \ (m,n+1) - \Delta u_n \ n^i(m,n) \quad , \qquad (2.37)$$

where

$$\Delta u_n = n^j(m,n) \ [\Delta u_*^j \ (m,n+1) - \frac{1}{4} \Delta u_+^j \ (m,n+2)] \qquad (2.38)$$

and Δu_*^i are the unadjusted components of the displacement increments one mesh spacing in from the clamped edge as obtained from the equations of motion.

The remainder of MOTION is concerned with energy calculations, the details of which are covered in Section 2.4. Briefly, the kinetic energy due to Δu_+^i is calculated and, if there is a pressure loading, the portion of the external work due Δu_+^i is also calculated. Then the total external work is computed and finally the energy dissipated by plasticity is determined.

Returning to the main program, see Figure 2.3, the next subroutine called is PDATA, which is described in Section 2.2. Following PDATA, subroutine DAMP is called in order to compute the energy removed by the damping, as described in Section 2.4. A check is next performed to determine if information for a restart should be collected by subroutine WRTAPE at this time step. If no restart information need be gathered, then the finite difference calculational loop is complete; otherwise, WRTAPE and PDATA are called, again completing the finite difference loop. This sequence of calculation has generated new values y^i, n^i, $\sigma^{\alpha\beta}$ and Δu_+^i of the fundamental variables from the old values y_-^i, n_-^i, $\sigma_-^{\alpha\beta}$ and Δu^i; the solution has been advanced a time step.

2.4 Energy Calculations

The energy calculations and the finite difference calculations are performed concurrently. The energy calculations use the results of the finite difference calculations to determine the current values of the kinetic energy, the strain energy, the external (pressure) work and the plastic work (i.e. the energy dissipated by plasticity). When the damping option is used, they also determine the energy removed by damping, called

the damping work. The energy calculations have no influence on the finite difference calculations and, hence, on the solution, except when the damping option is employed. This section presents the equations and procedure used in the REPSIL energy calculations; a theoretical report justifying the use of these will be forthcoming shortly.

Because the energy calculations are embedded in the finite different calculations, the flow chart, Figure 2.3, used in describing the latter calculation still pertains and will be referred to in the discussion that follows.

The first calculation performed is for the strain energy. Immediately after the current values of the mixed components of stress σ^{α}_{β} are computed in STRESS, Figure 2.5, the strain energy density ϕ per unit material coordinate volume at mesh point (m,n) and layer k is calculated using:

$$\phi = \frac{1}{2E}\left[(\sigma^1_1 + \sigma^2_2)^2 - 2\,(1+\nu)\quad(\sigma^1_1\,\sigma^2_2 - \sigma^1_2\,\sigma^2_1)\right] g^{\frac{1}{2}} \tag{2.39}$$

and simultaneously summed over all mesh points and layers. The strain energy calculation is completed in DGEOM, where the last sum is multiplied by the finite difference volume element giving the strain energy of the shell:

$$V = \sum_{m,n,k} \phi\ \Delta\eta^1 \Delta\eta^2 \Delta\zeta \quad . \tag{2.40}$$

The kinetic energy calculation is accomplished in subroutine KINET, which is , as shown in Figure 2.6, called by subroutine MOTION.

KINET The kinetic energy density ψ per unit material coordinate area of the middle surface at the mesh point (m,n) is determined:

$$\psi = \frac{1}{2}\,\Gamma_o\, a_o^{\frac{1}{2}}\ \frac{\Delta u^i_+}{\Delta t}\ \frac{\Delta u^i_+}{\Delta t} \quad . \tag{2.41}$$

Summing ψ over all mesh numbers (m,n) and multiplying by the finite difference area element, the kinetic energy of the shell is obtained:

$$T(t + \frac{1}{2}\,\Delta t) = \sum_{m,n} \psi\ \Delta\eta^1\ \Delta\eta^2 \quad . \tag{2.42}$$

Notice, since the displacement increments Δu^i_+ are for the time interval [t,t+Δt], the kinetic energy is properly centered at the time t + 1/2 Δt, as indicated. The kinetic energy at the

time t is determined by averaging the values of the kinetic energy at times $t + \frac{1}{2}\Delta t$ and $t - \frac{1}{2}\Delta t$:

$$T(t) = \frac{1}{2} [T(t + \frac{1}{2}\Delta t) + T(t - \frac{1}{2}\Delta t)] \quad . \qquad (2.43)$$

The external work calculation is done in two steps, accomplished by MOTION calling subroutine PWORK twice, see Figure 2.6.

PWORK This subroutine calculates the work ω per unit material coordinate area of the middle surface at the mesh point (m,n) due to the pressure P acting during half the displacement increment Δu^i using the equation

$$\omega = -\frac{1}{2} P \quad \Delta u^i n^i \quad , \qquad (2.44)$$

where the negative sign is a consequence of the pressure be oppositely directed to n^i. The subroutine then sums ω over the mesh points and returns to MOTION.

The first time PWORK uses the values of the displacement increments for the time interval $[t - \Delta t, t]$, determining the contribution to the external work of the interval $[t - 1/2\ \Delta t, t]$; the second time it uses the values for the increment $[t, t + \Delta t]$, determining the contributions of the interval $[t, t + 1/2\ \Delta t]$. These contributions are added and the result multiplied by the finite difference area element to give the total external work during the time interval $[t - 1/2\ \Delta t\ \ t + 1/2\ \Delta t]$:

$$\Delta W(t) = \sum_{m,n} [\omega(t - \frac{1}{4}\Delta t) + \omega(t + \frac{1}{4}\Delta t)]\ \Delta\eta^1\ \Delta\eta^2 \ ,$$

$$= - \sum_{m,n} [\ \frac{\Delta u^i + \Delta u^i_+}{2}\ n^i\ P^*]\ \Delta\eta^1\ \Delta\eta^2 \quad . \qquad (2.45)$$

This work increment, which is properly centered at time t, is then averaged with the work increment at time $t - \Delta t$ to give the work increment for the time interval $[t - \Delta t, t]$ centered at time $t - 1/2\ \Delta t$:

$$\Delta W(t - 1/2\ \Delta t) = 1/2\ [\Delta W(t - \Delta t) + \Delta W(t)]. \qquad (2.46)$$

This average work increment is added to the total external work up to the time $t - \Delta t$ to give the external work done up to the current time t:

$$W(t) = W(t - \Delta t) + \Delta W(t - 1/2\ \Delta t). \qquad (2.47)$$

The reasons for doing the external work calculation in this rather elaborate way are made clear in the forthcoming theoretical report on the energy calculations.

If the damping option is not in effect, then the only means by which energy is dissipated is through plastic flow. This unavailable energy is measured by the plastic work W_p, which is simply the difference of the total external work and the sum of the kinetic energy and strain energy:

$$W_p = W - T - V\ . \qquad (2.48)$$

The plastic work is computed in MOTION, just following the external work calculation, see Figure 2.6. With this calculation, the generation of current values of W, T and V is complete and the energy calculations end.

On the other hand if the damping option is in force, then there is an additional means of energy dissipation, measured by the damping work W_D. In this case the plastic work W_p is computed from

$$W_p = W - T - V - W_D\ . \qquad (2.49)$$

The damping work W_D is computed in subroutine DAMP which is called after MOTION, in which the above calculation for W_p is performed. Hence, for a proper sequencing of calculations, DAMP must compute the damping work up to the next time step $t + \Delta t$.

DAMP This subroutine, which is schematically summarized in Figure 2.7, controls the entire damping operations. As already mentioned in the introduction, these operations remove the kinetic energy of the system efficiently so that the shell approaches a static equilibrium configuration quickly. The kinetic energy is removed in two ways: first through viscous damping and second through the use of a kinetic energy annihilation (KEA) procedure. The KEA procedure is the principal means of energy removal, while the viscous damping mainly serves to smooth out disturbances in the solution caused by the abrupt nature of the KEA procedure. Conceptually, the KEA procedure involves "freezing" the position of the shell whenever the kinetic energy achieves a local maximum, so that the velocity and hence the kinetic energy vanish instantaneously, followed by an immediate "release" of the shell.

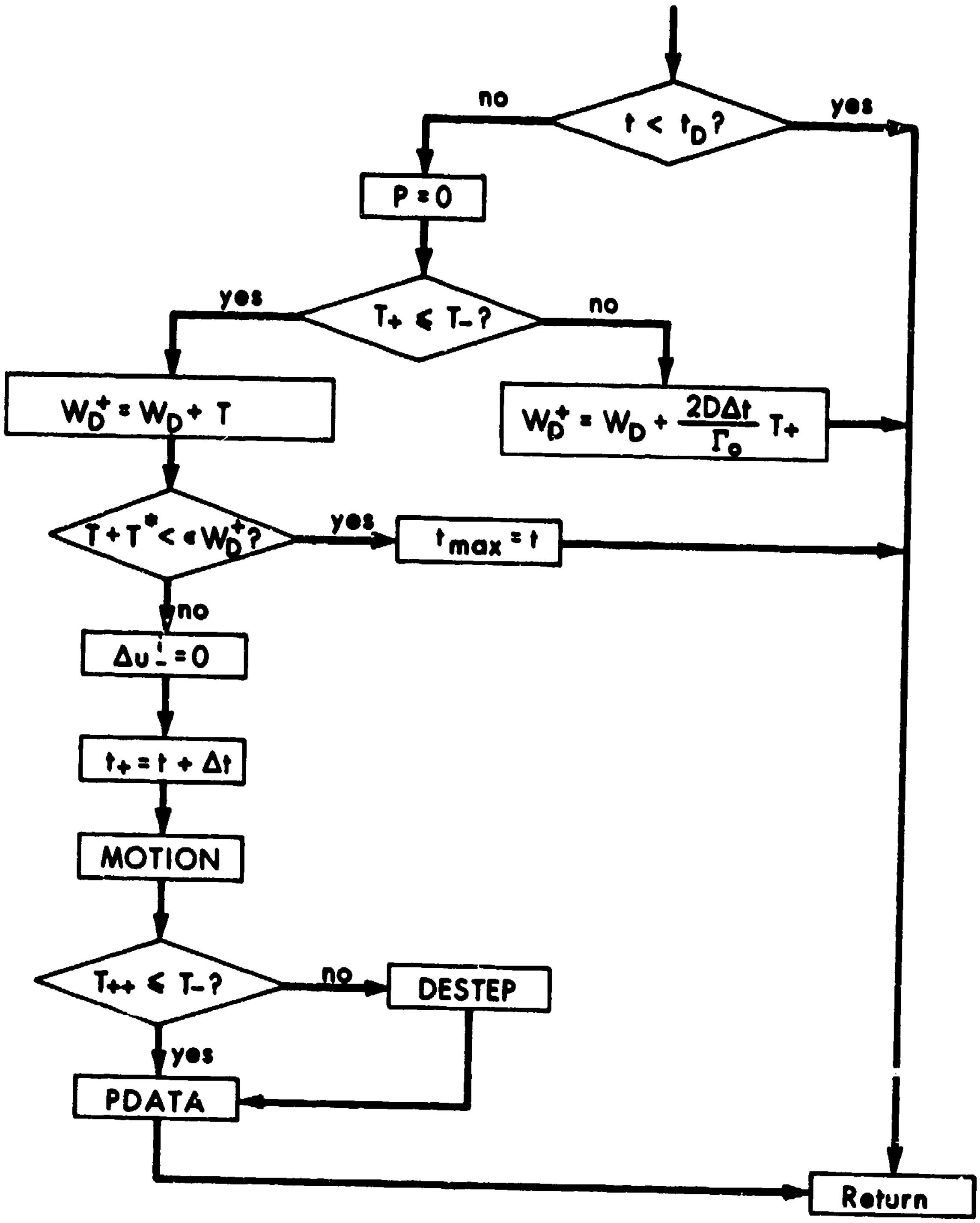


Figure 2.7 Flow Chart for Subroutine DAMP

Since it is highly unlikely that the shell will be "frozen" in a static equilibrium configuration, it will resume its motion on being "released", but with the total energy reduced by the amount of kinetic energy present when the maximum was achieved. This procedure is repeated at each maximum of the kinetic energy until the energy removed at some maximum is a small enough fraction of the total energy removed, at which time the shell is considered sufficiently close to its static equilibrium configuration and computations terminate*. This procedure is motivated by two considerations: first, the removal of energy is accomplished efficiently, since it occurs when the kinetic energy is at a peak and, second, should the kinetic energy be at its absolute maximum, then the corresponding configuration would be a static equilibrium configuration.

The subroutine begins by determining if the damping procedure is in force by checking the time t against the prescribed time t_D at which this procedure is to begin. If t_D is not exceeded the remainder of the subroutine is not used and calculations return to the main program. When t_D is exceeded, first the pressure P is set to zero. Second, the kinetic energy is checked as to whether it has just reached a local maximum by comparing its value T_+ at the time $t + 1/2\ \Delta t$ to its value T_- at $t - 1/2\ \Delta t$.** If $T_+ > T_-$ then no maximum has been reached and the subroutine adds the increase in the damping work due to viscous damping over this interval, which is simply a linear function of T_+, to the damping work W_D at the time t to obtain the damping work W_D at the time $t + \Delta t$:

$$W_D^+ = W_D + \frac{2\ D\ \Delta t}{\Gamma_0} T_+ \quad . \qquad (2.50)$$

On the other hand, if $T_+ \leq T_-$, a local maximum has been reached and the KEA procedure goes into effect. To maintain the energy balance, the kinetic energy removed at this time t is added to the damping work:

$$W_D^+ = W_D + T. \qquad (2.51)$$

---

* This method for reducing kinetic energy appears to be in common use; See, for example, DAHL, BEELER and BOURQUIN [4] who use this method to obtain computer solutions of some solid state physics problems.

** Cf. the description of subroutine KINET, where the notation $T(t+1/2\ \Delta t)$ and $T(t-1/2\ \Delta t)$ was used for T_+ and T_-.

The sum of kinetic energy T removed at this time and the kinetic energy T* removed by the previous KEA calculation is compared to the damping work W_D^+. If this sum is a small enough fraction of W_D^+, then the maximum time for the problem t_{max} is set equal to the present time t, after which the main program causes the problem to terminate, see Figure 2.3. If the sum is not sufficiently small, then the displacement increments Δu_+^i are set equal to zero, making the velocities at time t + 1/2 Δt and, hence, the kinetic energy T_+ vanish. Since $\Delta u_+^i = 0$, the position of the shell and stress field at the time t + Δt will remain the same as at the previous time t. Hence, these variables need not be recalculated and the subroutine can proceed to increase the time step and call MOTION in order to calculate the displacement increments Δu_{++}^i for the interval [t + Δt, t + 2 Δt] and the kinetic energy T_{++} at the time t + 3/2 Δt. The kinetic energy T_{++} is next compared with T_-. If $T_{++} \leq T_-$ then the subroutine finishes the damping operations by calling PDATA in order to collect some plotting information and returns to the main program. However, if $T_{++} > T_-$ then experience has shown that a numerical instability due to the KEA operations is likely to occur. To remedy this the subroutine calls subroutine DESTEP in order to compute a smaller stable time increment Δt*.

DESTEP This subroutine calculates a decreased time step Δt* that prevents the KEA operations from causing an instability:

$$\Delta t^* = \left(\frac{T_-}{T_{++}}\right)^{\frac{1}{2}} \Delta t \,. \tag{2.52}$$

Then the values of Δu^i and T_{++} are adjusted for the decreased time step by scaling:

$$\Delta u_{++}^{*i} = \frac{2\,\Gamma_o + D\Delta t}{2\,\Gamma_o + D\Delta t^*} \left(\frac{\Delta t^*}{\Delta t}\right)^2 \Delta u_{++}^i \,, \tag{2.53}$$

$$T_{++}^* = \left(\frac{2\,\Gamma_o + D\Delta t}{2\,\Gamma_o + D\Delta t}\right)^2 \left(\frac{\Delta t^*}{\Delta t}\right)^2 T_{++} \,. \tag{2.54}$$

Returning from DESTEP back to DAMP, PDATA is called and the calculations return to the main program as before.

2.5 Surface Strain Calculations

The surface strain calculations, like the energy calculations, are embedded in the finite difference sequence of calculations. They do not influence the solution, but their results are a consequence of the solution and provide useful local measures of the validity and reliability of the solution. The elongational strains in prescribed directions at prescribed locations on the bounding surfaces of the shell are calculated. The equations used are derived in [1; App. D] and give the exact elongation per unit initial length, with no approximations based on the smallness of strain being invoked. The strains are intended to simulate the readings of strain gages bonded to the shell at these locations.

The strain calculations take place in subroutine STRAIN. However, before these calculations are performed, the interpolation coefficients and the mesh numbers bracketing the strain locations are calculated in START, see Figure 2.2, and subroutine ABINIT is called during the initial pass through DGEOM, see Figure 2.4.

ABINIT This subroutine uses the mesh numbers of the mesh points bracketing the strain locations, as determined to START, to select the initial values of $a_{\alpha\beta}$ and $b_{\alpha\beta}$ at the bracketing mesh points. These values are stored in arrays for later use in STRAIN.

The above calculations are performed initially and only once. The covariant components of strain $\varepsilon_{\alpha\beta}$ on the bounding surfaces of shell are computed in DGEOM each time step for every mesh point using the equation

$$\varepsilon_{\alpha\beta} = \varepsilon^{-}_{\alpha\beta} + \frac{1}{2} (\Delta a_{\alpha\beta} \pm h \, \Delta b_{\alpha\beta}), \qquad (2.55)$$

where $\varepsilon^{-}_{\alpha\beta}$ are the covariant strain components at the previous time $(t - \Delta t)$, h is the shell thickness and the + or - sign depends on whether the bounding surface is on the negative or positive side of the normal n^i, respectively. The covariant components of strain are used in STRAIN, immediately following DGEOM.

STRAIN This subroutine calculates the elongational strains in predetermined directions at predetermined locations on the surfaces of the shell. It also computes the components of the total displacement of a predetermined locations on the middle surface. These directions and locations are specified in the input data, see Section 3.2. At each strain location four elongational strains are found: two along the coordinate curves and two in directions specified in the input data by the angles θ made with the η^1 coordinate curve, see Figure 3.5. The first time

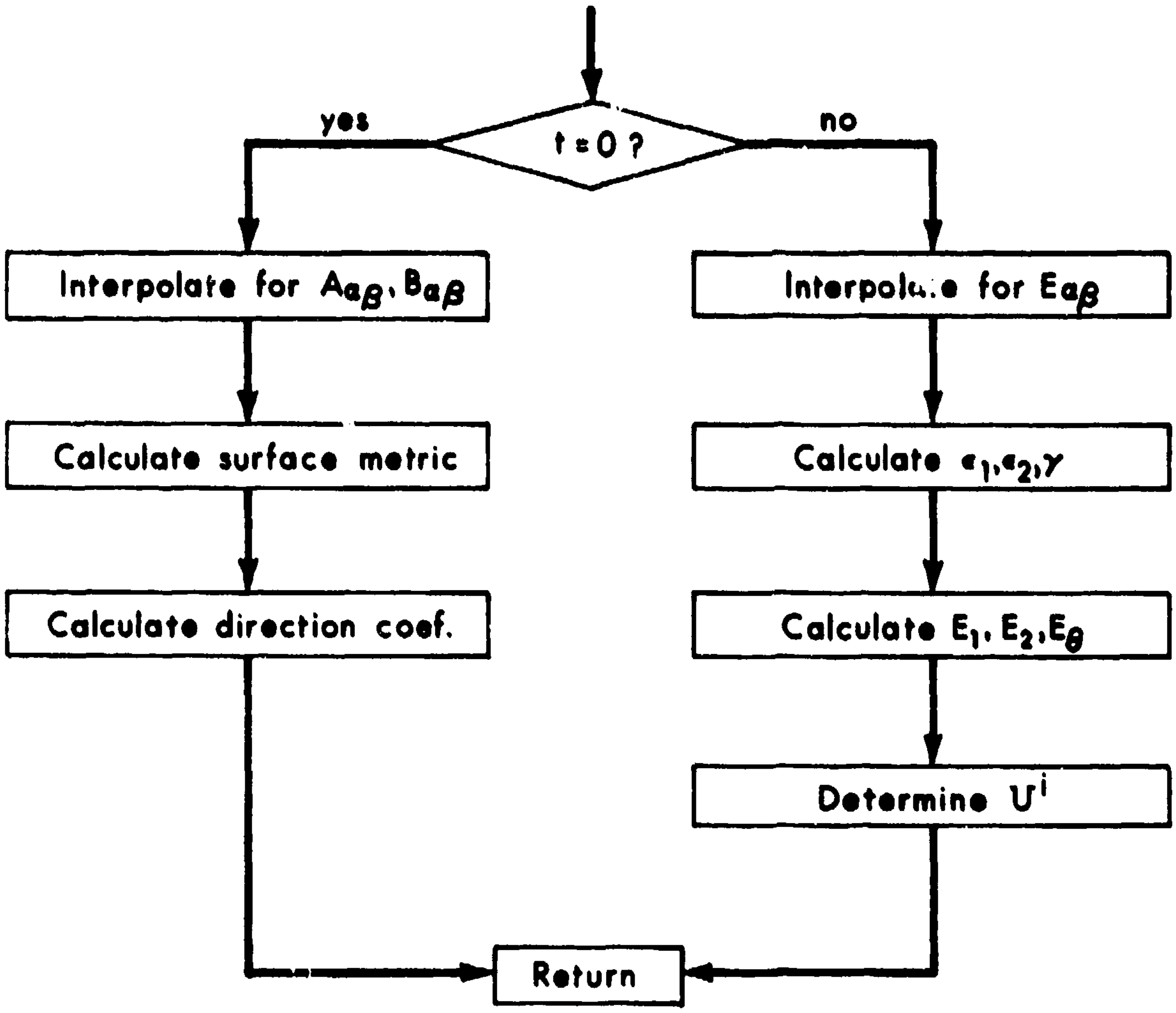


Figure 2.8 Flow Chart for Subroutine STRAIN

this subroutine is executed, see Figure 2.8, the interpolation coefficients calculated in START are used to linearly interpolate the initial values of $a_{\alpha\beta}$ and $b_{\alpha\beta}$ to obtain the corresponding values $A_{\alpha\beta}$ and $B_{\alpha\beta}$ at the strain locations. From these values, the components of the surface metric $G_{\alpha\beta}$ at these locations are calculated:

$$G_{\alpha\beta} = A_{\alpha\beta} \pm h\, B_{\alpha\beta} \quad , \tag{2.56}$$

where, as in (2.55), + or - depends on the surface lying in the negative or positive direction of n^1. Also, the direction coefficients α and β are calculated from the angle θ specifying the strain directions:

$$\alpha = \sin\theta \;/\; \sqrt{1 - \delta^2} \qquad \beta = \cos\theta - \alpha\delta \tag{2.57}$$

where δ is a function of $G_{\alpha\beta}$:

$$\delta = G_{12} \;/\; \sqrt{G_{11}G_{22}} \tag{2.58}$$

This calculation ends the initial pass through the subroutine. For all subsequent time steps, the subroutine uses a different computational loop to calculate the elongational strains and the components of total displacement, as shown in Figure 2.8. First, the values $\varepsilon_{\alpha\beta}$ at the mesh points bracketing the strain locations are linearly interpolated to give the covariant components of strain $E_{\alpha\beta}$ at the strain locations. These components are combined with $G_{\alpha\beta}$ to give the intermediate strain components

$$\varepsilon_1 = E_{11}/G_{11} \;,\; \varepsilon_2 = E_{22}/G_{22} \;,\; \gamma = E_{12} \;/\; \sqrt{G_{11}G_{22}} \quad . \tag{2.59}$$

From these intermediate components, the subroutine determines the elongational strains along the η^1 and η^2 coordinate curves:

$$E_1 = \sqrt{1 + 2\,\varepsilon_1} - 1 \quad , \quad E_2 = \sqrt{1 + 2\,\varepsilon_1} - 1 \tag{2.60}$$

and in the direction specified by θ:

$$E_\theta = \sqrt{1 + 2\,(\beta^2\varepsilon_1 + 2\,\alpha\beta\gamma + \alpha^2\,\varepsilon_2)} - 1 \qquad (2.61)$$

The remainder of the subroutine involves determining the total displacement components U^i at a predetermined location. The components of displacement increments Δu^i at the mesh point surrounding the displacement location are linearly interpolated to give the components of displacement increment ΔU^i at this location. These components are added to the previous values of the components of the total displacement U^i to give their current values.

$$U^i = U^i_- + \Delta U^i \quad . \qquad (2.62)$$

3. DESCRIPTION OF INPUT

3.1 Input Cards

The data needed to run REPSIL are supplied on input cards and, in the case of certain pressure loadings, on a user-generated input tape. Instructions for generating the pressure input tape are given in Section 6.2. The input cards assign values to the FORTRAN variables listed in Table 3.1 in that order using the formats indicated.

Table 3.1 List of Input Cards

CARD	VARIABLES	FORMAT
1	TITLE	10A8
2	MESH, NMESH, LAYER, YLDFAC	3I5,E12.6
3	MAXC, NCONT, NRITE, DELTAT	3I5,E12.6
4	IBEC1, IBEC2, IBEC3, IBEC4	4I5
5	LOAD, LPRESS, MDAMP, DAMPF, DFACT	3I5,2E12.6
6	E, FNU, SIGZ, RHO, THICKN, NSFL, ISR	5E12.6,2I5
7	(SSIG(J), SEPS(J), DSR(J), PSR(J), J=1, NSFL)	4E15.7
8	NPRINT, (JCHK(J), J=1,3)	4I5
9	NUMCY, (NCYCH(J), J=1, NUMCY)	16I5
10	NLPRIN, (JCYNLP(J), J=1,NLPRIN)	16I5
11	N3D (NC3DP(J), J=1, N3D)	16I5
12	ETAD1, ETAD2, NSTRN	2E10.4,I5
13	(ETAG1(I), ETAG2(I), ANGLE(I), ANGLB(I), NETAG(I), I=1, NSTRN)	4E10.4,I5
14	LENGTH, WIDTH [for flat plate]	2E12.6
	LENGTH, RADIUS, THETA [for cylindrical shell]	3E12.6
	LENGTH, RADI, RADF, THETA, MASH [for conical shell]	4E12.6,I5
15	MI, MF, NI, NF, VR, NV	4I5,E12.6,I5
16	M,N,V	2I5,E12.6

The following rules should be obeyed in preparing the input cards.

- Omit card 7 if either NSFL = 0 or NSFL = 1 and ISR = 0; otherwise, the number of card 7's must match NSFL.
- The number of card 13's must match NSTRAN $\geq$ 1.
- Only one card 14 is used, with the data matching the particular subroutine INGEOM used.
- Omit cards 15 and 16 if LOAD = 1 or NCONT > 0.
- Omit card 16 if NV = 0; otherwise, the number of card 16's must equal NV.

The input cards can be grouped according to the type of data they supply:

Cards 2,3	Data controlling the finite difference and numerical analysis.
Card 4	Parameters for selecting boundary conditions.
Card 5	Data controlling the type of loading and the damping option.
Card 6,7	Material properties.
Card 8,9,10,11	Printing and plotting control numbers.
Cards 12,13	Data specifying locations where displacement components and surface strains are to be calculated.
Card 14	Dimensions of shell.
Card 15,16	Data characterizing the initial impulse velocity.

3.2 Description of Input Variables

The input variables are described below in the order in which they appear on the input cards, as listed in Table 3.1. The dimensions of a variable are indicated by capital letters in square brackets following the short underlined description of the variable, with F representing force, L length, and T time. The program is written to accept any consistant set of dimension units. For example, the mass density in the pound-inch-second system of units for a material weighing 1 pound per cubic inch would be $\frac{1}{386}\ \frac{\text{lb-sec}^2}{\text{in}^4}$.

Card 1 TITLE Title to identify run. Not to exceed 80 alphanumeric characters.

Card 2 MESH Number of mesh intervals in the η^1 direction.

NMESH Number of mesh intervals in the η^2 direction.

Figures 3.6 - 3.8 show the orientation of the mesh relative to the η^1 and η^2 directions for the initial geometries presently programmed in REPSIL. The choice of MESH and NMESH should only be based on the portion of the shell to be actually analyzed; additional intervals due to exterior mesh points along symmetry boundaries should be disregarded. MESH and NMESH are limited by the maximum number of M and N mesh elements permitted by DIMENSION and COMMON statements (see Section 3.3):

$$\text{MESH} \leq M_{max} - \begin{cases} 2 \ ; \ \text{IBCE3} = 1,3 \\ 3 \ ; \ \text{IBCE3} = 2 \end{cases}$$

$$\text{NMESH} \leq N_{max} - \begin{cases} 1 \ ; \ \text{IBCE2} = 1,3 \\ 2 \ ; \ \text{IBCE2} = 2 \end{cases}$$

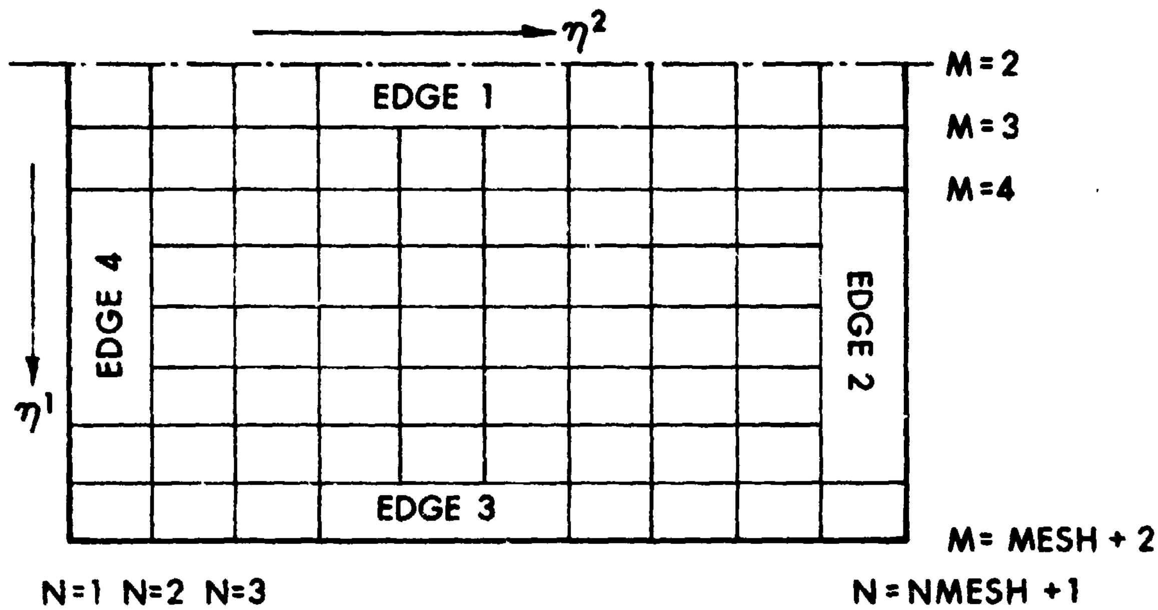
LAYER Number of layers into which the shell thickness is divided. The shell is divided into layers, within which the stress is assumed constant, in order to facilitate the modelling through the thickness of the stress profile resulting from plasticity. Hence, the greater the number of layers used, the more accurately is the stress profile presumably modelled, but at the expense of longer computation times and greater memory requirements. LAYER = 4 has been found to be a good compromise giving reasonably accurate deflections.

YLDFAC Parameter controlling the "thickness" of ellipsoidal annuli surrounding yield surface in stress space. The ellipsoidal annuli divided the excursions of the stress increment outside yield surface into subincrements making the calculation of the stress on the yield surface more accurate, see Appendix B. Accuracy increases with value of YLDFAC, but at the expense of increased computation times, with YLDFAC = 1 a good compromise. In order not to use this option set YLDFAC = 0.

Card 3	MAXC	Time step at which it is desired to terminate the problem.
	NCONT	Time step at which it is desired to begin the problem. For new problems NCONT = 0 and for restart problems NCONT = time step from which solution is continued. Notice that always NCONT < MAXC.
	NRITE	Time steps elapsed between the gathering of restart data. Restart information written on the restart tape(tape #1)every NRITE time steps. If it is planned to restart a problem, NRITE $\leq$ MAXC; if restart information is not desired, make NRITE > MAXC and omit tape.
	DELTAT	Finite difference time increment [T]. Using equations (2.3), the program calculates time increments that assure numerical stability in the membrane and bending modes of vibration and then chooses the minimum of these and the input DELTAT to use in the finite difference calculations. If DELTAT = 0.0, the program chooses the minimum stable time increment.
Card 4	IBEC1 IBEC2 IBEC3 IBEC4	Number prescribing boundary conditions along the edges of the shell. In Figure 3.1 the edges are inumerated relative to the (M,N) grid and the admissible values of the boundary control numbers at each edge are listed.

Clamped edge condition (1). Coordinates of middle surface y^i and components of normal n^i are fixed at their initial values along this edge.

Symmetry edge condition (2). Edge lies in a symmetry plane about which the shell and the loads are symmetrically distributed. Edge 1 is always a symmetry edge located in the $y^1 = 0$ symmetry plane, see Figures 3.6 - 3.8. The symmetry plane for edge 2 is y^2 = LENGTH, see Figure 3.6 and 3.7. The symmetry plane for edge 3 is the same as that for edge 1, namely $y^1 = 0$, and hence is applicable to shell intersecting this plane twice, such as cylinders and cones, see Figure 3.7 and 3.8. Care should be taken that the symmetry edge condition be compatible with the particular shell geometry treated, e.g. although IBCE3 =2 is admissible, it is certainly not appropriate to the flat plate (Figure 3.6) or the cylindrical panel with THETA < π (Figure 3.7).



EDGE 1: IBCE1 = 2 (SYMMETRY)

EDGE 2: IBCE2 = 1 (CLAMPED), 2 (SYMMETRY) OR 3 (HINGED)

EDGE 3: IBCE3 = 1 (CLAMPED), 2 (SYMMETRY) OR 3 (HINGED)

EDGE 4: IBCE4 = 1 (CLAMPED) OR 3 (HINGED)

Figure 3.1 Admissible Boundary Conditions

Hinged edge conditions (3). Coordinates of middle surface y^i are fixed at their initial values along this edge, but normal n^i is free to rotate about edge.

Card 5 LOAD — Number controlling mode of loading

= -1, initial impulse velocity and pressure-time loading,

= 0, initial impulse velocity,

= 1, pressure-time loading.

Initial velocity distribution, other than those representable by cards 15 and 16, and pressure-time histories must be supplied by the user in appropriate form through subroutines INVEL and PRESS; instructions for doing this are in Sections 6.2 and 6.3, respectively.

LPRESS — Distribution of pressures over shell after time steps LPRESS fixed at the LPRESS distribution. If LOAD = 0, set LPRESS = 0. If user does not desire to fix pressure distribution, make LPRESS > MAXC.

MDAMP — Time step at which damping starts. Numerical damping is used to rapidly slow down the motion of the shell in order to obtain a final equilibrium configuration. MDAMP is selected after most of the plastic dissipation is over. This entails a preliminary run in order to estimate from the energy balance when plastic deformation is vitually finished; the damping run is continued from the time step closest to MDAMP as a restart problem (see Section 3.4). If damping is not desired set MDAMP > MAXC, otherwise values for DAMPF and DFACT must be supplied below.

DAMPF — Viscous damping coefficient used in smoothing solution during damping [FT/L^3]. Should not be too large in order to avoid overdamping and consequently prolonging the time to reach a final configuration.

DFACT — Parameter controlling termination of problem during damping. If the ratio of the sum of the energies removed in two consecutive kinetic energy annihilations to the damping work is less than DFACT, the problem terminates (see Figure 2.7). The smaller DFACT is made, the less the residual kinetic energy at termination, but at the expense of longer machine times.

Card 6 E Young's modulus $[F/L^2]$.

FNU Poisson's ratio.

SIGZ Yield stress $[F/L^2]$. For perfectly plastic behavior SIGZ is the maximum stress σ_0 on the uniaxial loading curve, Figure 3.2; for strain hardening behavior SIGZ is the stress σ_1 at the first change in slope in the polygonal approximation to the loading curve, Figure 3.3.

RHO Initial mass density per unit volume $[FT^2/L^4]$.

THICKN Thickness of shell [L].

NSFL Number of changes in slope in the polygonal approximations to the uniaxial loading curve (equal to the number of stress sublayers):

= 0, no plasticity ≡ elastic behavior,

= 1, elastoplastic with no strain hardening,
> 1, elastoplastic with strain hardening.

ISR Strain rate sensitivity control

= 0, plasticity is strain rate independent,

= 1, plasticity is strain rate dependent.

Card 7* SSIG(J),SEPS(J) Stress $[F/L^2]$ and strain [L/L] at points of slope change of the polygonal approximation to the uniaxial loading curve, Figure 3.3, where J = 1, NSFL. The program automatically makes these data compatible with those on Card 6 by setting SSIG(1) = SIGZ and SEPS(1) = SIGZ/E. For the strain rate sensitive case, take these data from the static loading curve.

DSR(J),PSR(J) Emperical constants used to model strain rate sensitive behavior, d and p of equation (2.21). Pair of constants must be specified for each slope change (i.e. each stress sublayer) on the polygonal approximation to the loading curve, J = 1, NSFL. On the stress-strain diagrams, Figure 3.4, the straight line portions of the

* Omit card 7 if either NSFL = 0 or NSFL = 1 and ISR = 0.

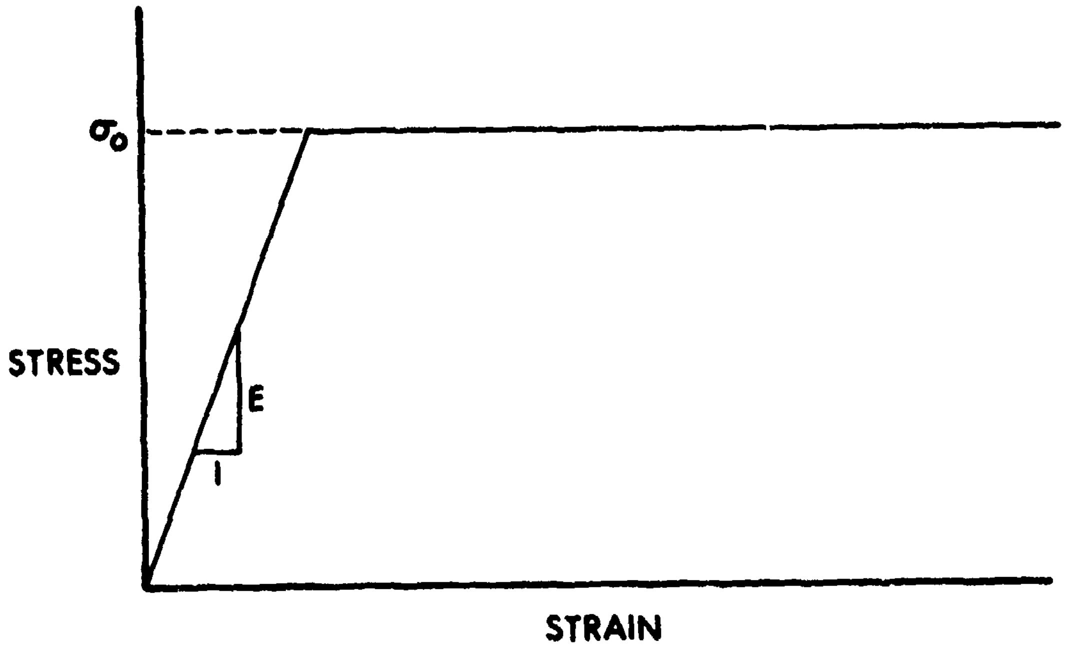


Figure 3.2 Uniaxial Loading Curve for the Elastic/Perfectly-Plastic Constitutive Model

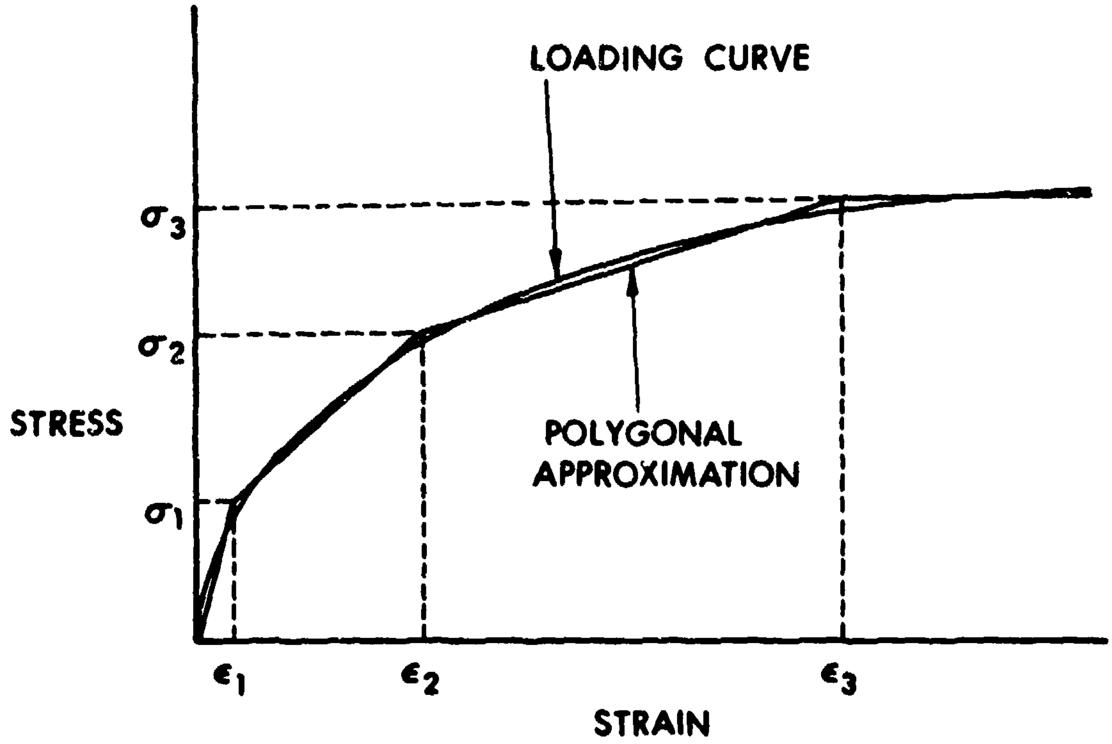
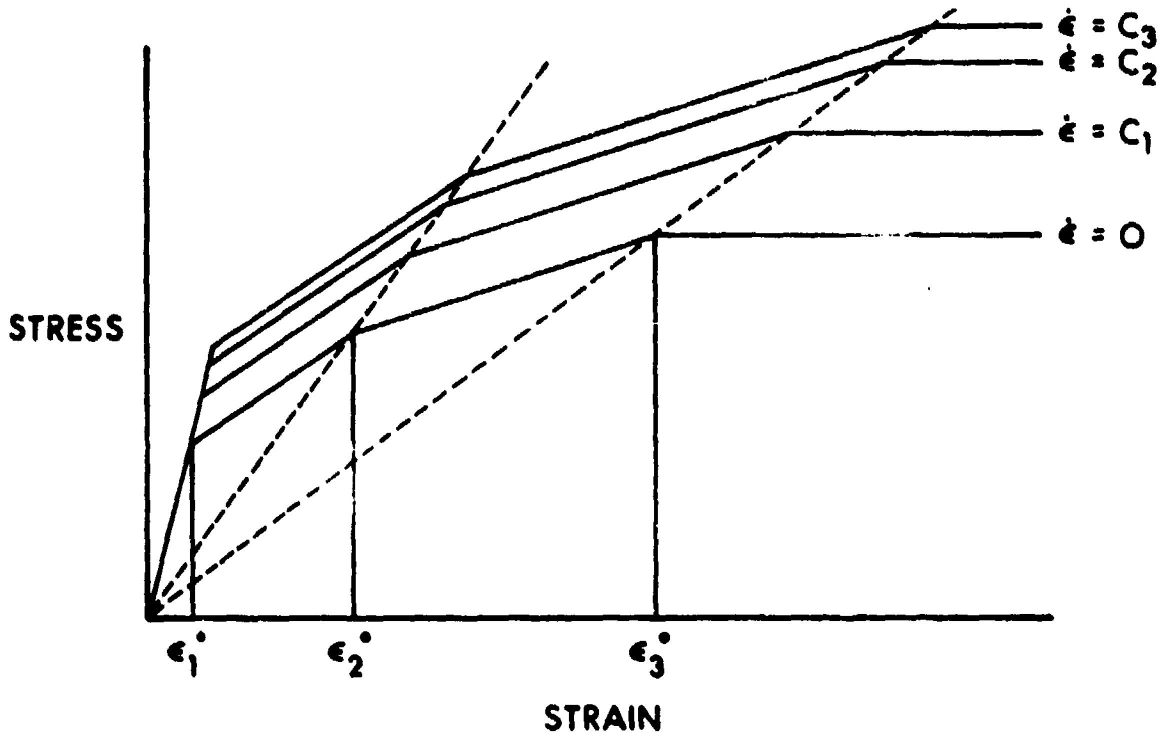
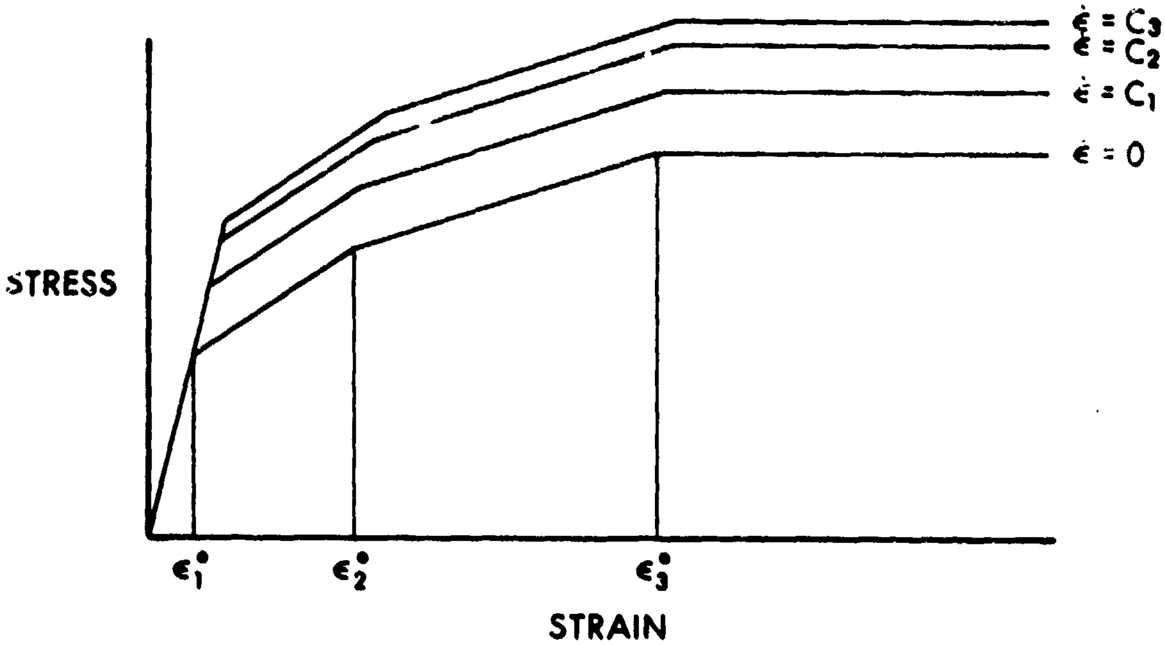


Figure 3.3 Uniaxial Loading Curve for the Strain Hardening Constitutive Model and Polygonal Approximation to Loading Curve



a. Strain rate parameters the same at each point of slope change.



b. Strain rate parameters differ at each point of slope change.

Figure 3.4 Polygonal Approximations to the Uniaxial Loading Curve at Various Constant Strain Rate Levels for Strain Rate Sensitive Materials

constant strain rate curves are parallel to the corresponding portions of the static loading curve. The strain ε_j at each point of slope change is magnified from the corresponding static strain ε_j° by the rate sensitivity factor:

$$\varepsilon_j = \varepsilon_j^\circ \left[1 + \left(\frac{\dot{\varepsilon}}{d_j}\right)^{\frac{1}{p_j}}\right] ,$$

where $\dot{\varepsilon}$ is the strain rate, see (2.22). If the same values of DSR(J) and PSR(J) are used for all J, the stress σ_j at each point of slope change is also magnified from the corresponding static stress σ_j°:

$$\sigma_j = \sigma_j^\circ \left[1 + \left(\frac{\dot{\varepsilon}}{d_j}\right)^{\frac{1}{p_j}}\right] ,$$

so that

$$\sigma_j = (\sigma_j^\circ / \varepsilon_j^\circ)\, \varepsilon_j ,$$

as illustrated in Figure 3.4a.

Card 8 NPRINT <u>Number of elapsed time steps between surface strain prints</u>. This print is described in Section 4.1.5. The remainder of the strain print data is specified on Card 13. If this print is not desired, set NPRINT > MAXC.

JCHK(J) <u>Numbers controlling the printing of output data</u>:

JCHK(1), components of displacement increments,

JCHK(2), coordinates of middle surface and the pressure,

JCHK(3), components of surface normal;

$$\text{JCHK(J)} = \begin{cases} 0, & \text{data not printed,} \\ 1, & \text{data printed.} \end{cases}$$

Card 9 NUMCY <u>Number of time steps for which JCHK(J) controlled data and energy balance data are to be printed</u>. Sections 4.1.2 and 4.1.3 describe these prints.

NCYCH(J) Time steps at which JCHK(J) controlled data and energy balance data are printed. If these prints are not desired, set NUMCY = 1 and NCYCH(1) > MAXC.

Card 10 NLPRIN Number of time steps for which the LMAT (M,N,K) array is printed. See Section 4.1.4 for a description of this print.

JCYNLP(J) Time steps at which the LMAT (M,N,K) array is printed If this print is not desired, set NLPRIN = 1 and JCYNLP(1) > MAXC.

Card 11 N3D Number of time steps for which isometric and cross-sectional plots are drawn. A description of the plotting capabilities of REPSIL is given in Section 4.2.

NC3DP(J) Time steps at which isometric and cross-sectional plots are drawn. If plots are not desired, set N3D=1 and Nc3DP (1) > MAXC.

Card 12 ETADI,ETAD2 Material coordinates of location at which the components of displacements are calculated and plotted [dimensions correspond to those for ETAG1(I), ETAG2(I) below].

NSTRN Number of locations at which surface strains are calculated and plotted.

Card 13 ETAG1(I), ETAG2(I) Material coordinates of locations at which surface strains are calculated and plotted, Figure 3.5. Dimensions depend on subroutine INGEOM:

Flat plate, distance along width and length, Figure 3.6,

Cylinder, angle in degrees from the symmetry plane and distance along axis, Figures 3.7.

Cone, angle in degrees from the symmetry plane and arclength along the cone generator, Figure 3.8.

ANGLE(I),ANGLB(I) Angles θ measured in degrees from the η^1 direction counterclockwise about the normal of the directions in which surface strains are calculated, Figure 3.5, where $0 \leq \theta < 180°$.

NETAG(I) Number selecting the bounding surface on which surface strain calculations are performed, Figure 3.5:

= 0, surface on positive side of normal,

= 1, surface on negative side of normal.

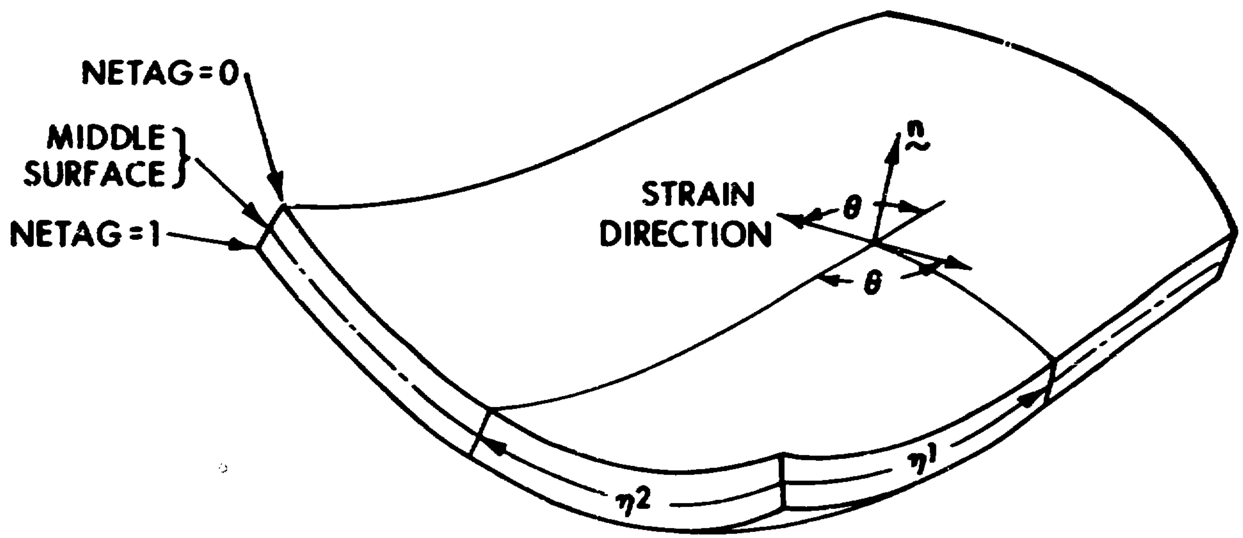


Figure 3.5 Material Coordinates η^{α} Locating Surface Strains Position and Orientation of Strain Direction Relative to the η^{1} Direction

Card 14 — Dimensions of shell specified on this card. Data specifying dimensions must be compatible with the particular subroutine INGEOM used. The next three figures show the orientations relative to the Cartesian coordinate axes y^i and the associated compatable boundary conditions for the three INGEOM subroutines presently programmed in REPSIL.

LENGTH — Length of plate along symmetry boundary [L], Figure 3.6. Length of cylinder axis [L], Figure 3.7. Length of cone along axis [L], Figure 3.8.

WIDTH — Width of plate up to symmetry boundary [L], Figure 3.6.

RADIUS — Radius of cylinder [L], Figure 3.7.

RADI,RADF — Small and large radii of cone [L], Figure 3.8.

THETA — Angle subtended by cylindrical or conical panel measured from the symmetry plane [Degrees], Figures 3.7 and 3.8.

MASH — Number controlling mesh proportions for cone:

= 0, equal mesh intervals along meridian, Figure 6.2a,

= 1, constant mesh proportions, Figure 6.2b.

For the details of this option see Section 6.4.

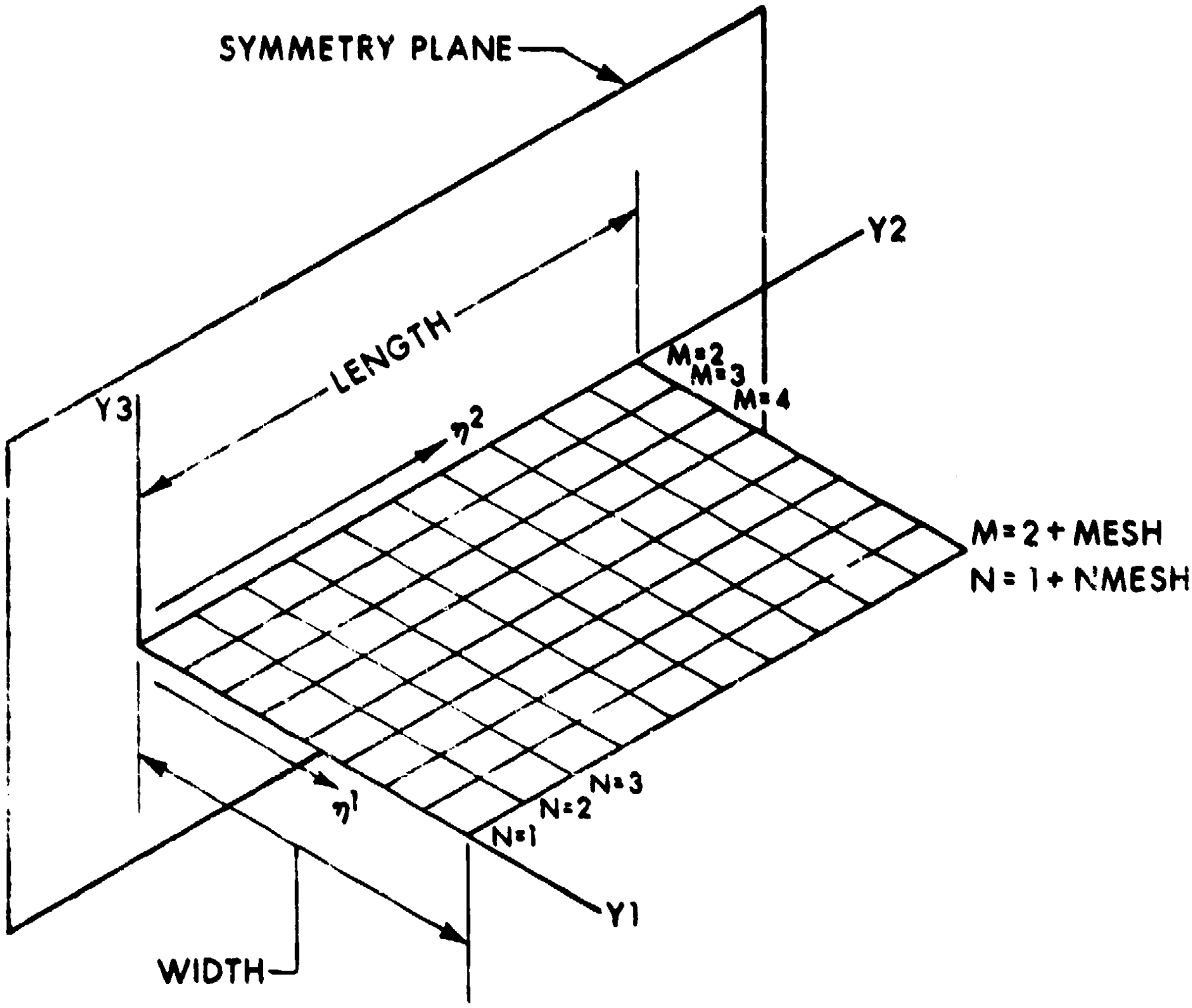
Card 15* — This card gives the data on the uniform initial impulse velocity field and gives the number of points with nonuniform impulse velocity, see Figure 3.9. As indicated in the figure, velocities are directed in the opposite sense to the normal.

MI,MF — Minimum and maximum values of mesh number M for point receiving uniform initial impulse velocity VR. $2 \leq MI < MF \leq MESH + 2$.

NI,NF — Minimum and maximum values of mesh number N for point receiving uniform initial impulse velocity VR. $1 \leq NI < NF \leq NMESH + 1$.

---

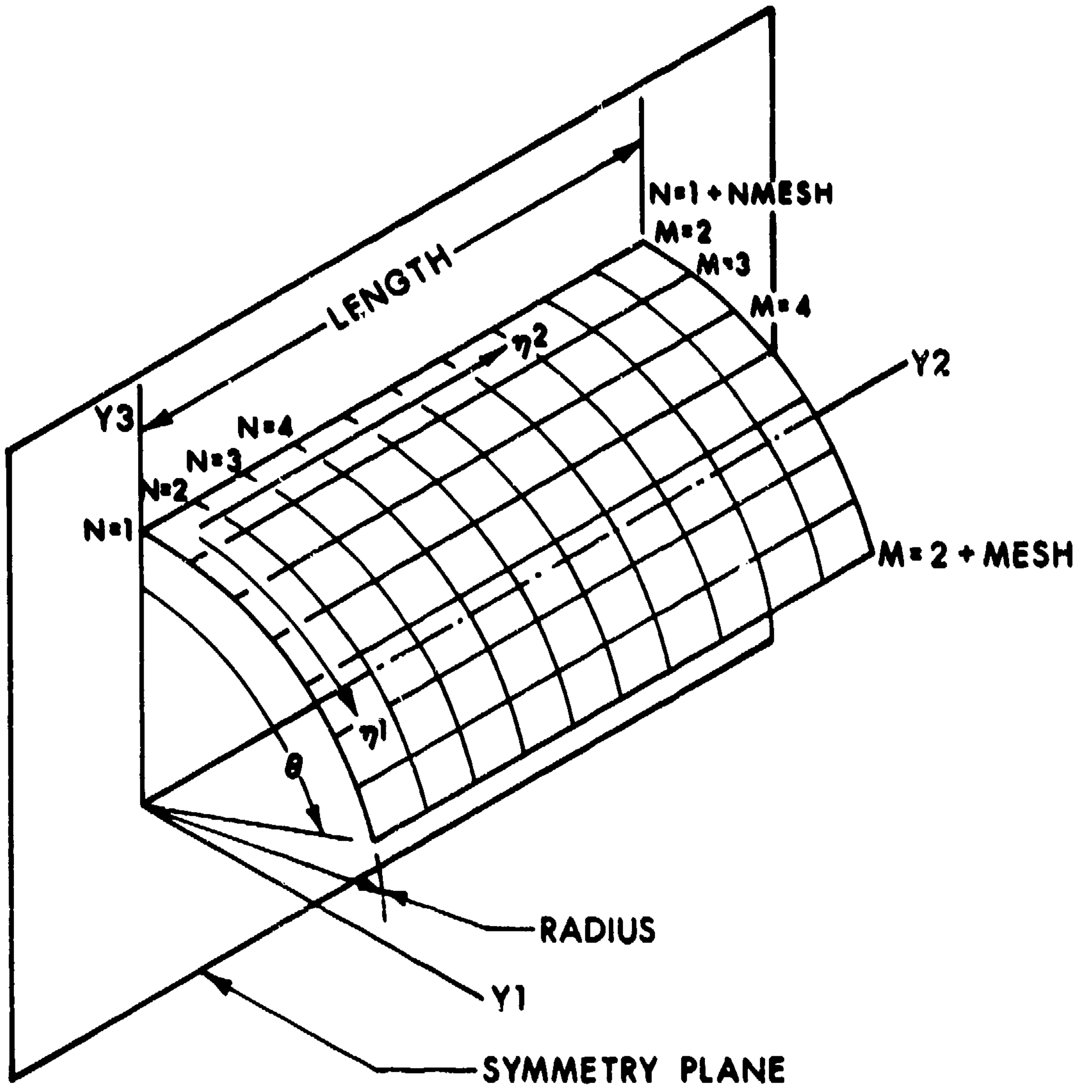
* Cards 15 and 16 omitted if LOAD = 1 or NCONT > 0.



Boundary conditions compatible with flat plate geometry (see Figure 3.1 for key):

IBCE1 = 2
IBCE2 = 1, 2 or 3
IBCE3 = 1 or 3
IBCE4 = 1 or 3

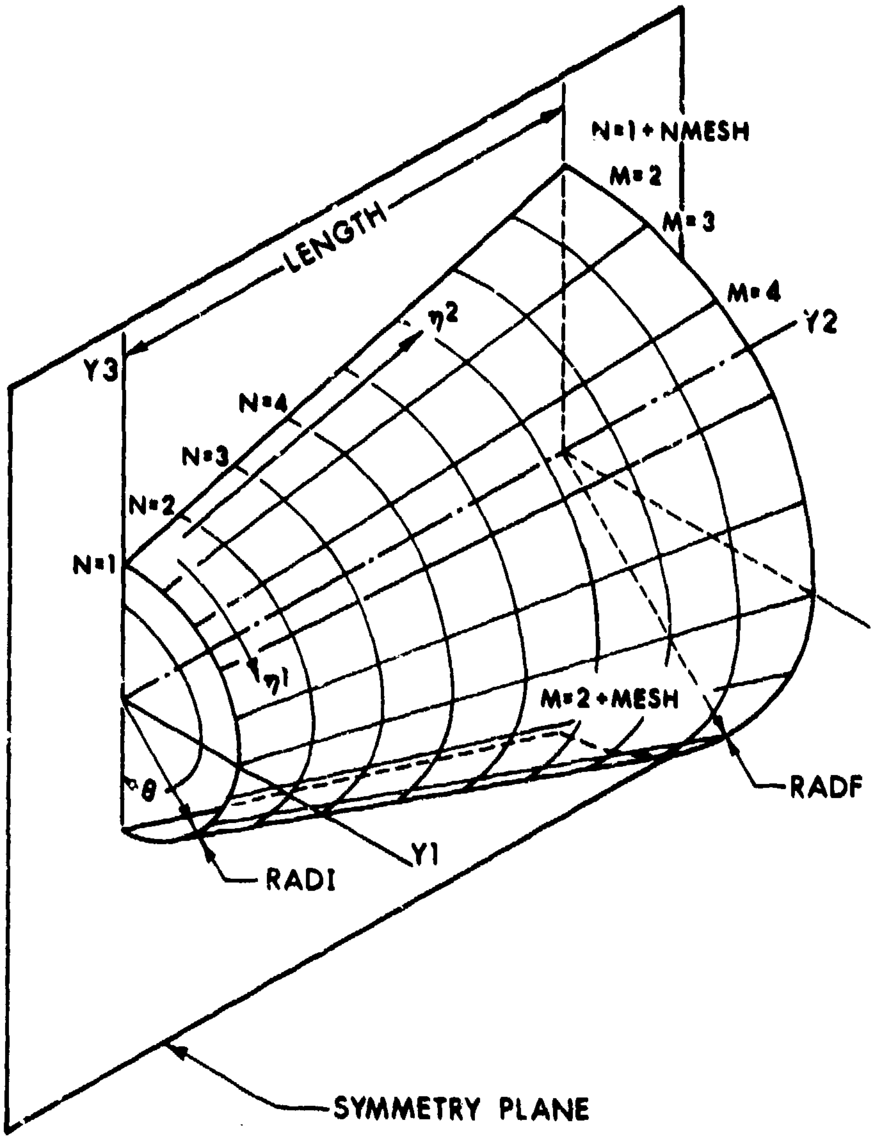
Figure 3.6 Flat Plate Geometry



Boundary conditions compatible with cylindrical geometry (see Figure 3.1 for key):

IBCE1 = 2
IBCE2 = 1, 2 or 3
IBCE3 = 1 or 3 when $0 < \sigma < 180°$ (as shown)
= 1, 2 or 3 when $\sigma = 180°$
IBCE4 = 1 or 3

Figure 3.7 Cylindrical Shell Geometry



Boundary conditions compatible with conical geometry (see Figure 3.1 for key):

IBCE3 = 2
IBCE2 = 1 or 3*
IBCE3 = 1 or 3 when 0 < θ < 180°
= 1,2 or 3 when θ = 180° (as shown)
IBCE4 = 1 or 3

*IBCE2 = 2 would result in discontinous slope across boundary .

Figure 3.8 Conical Shell Geometry

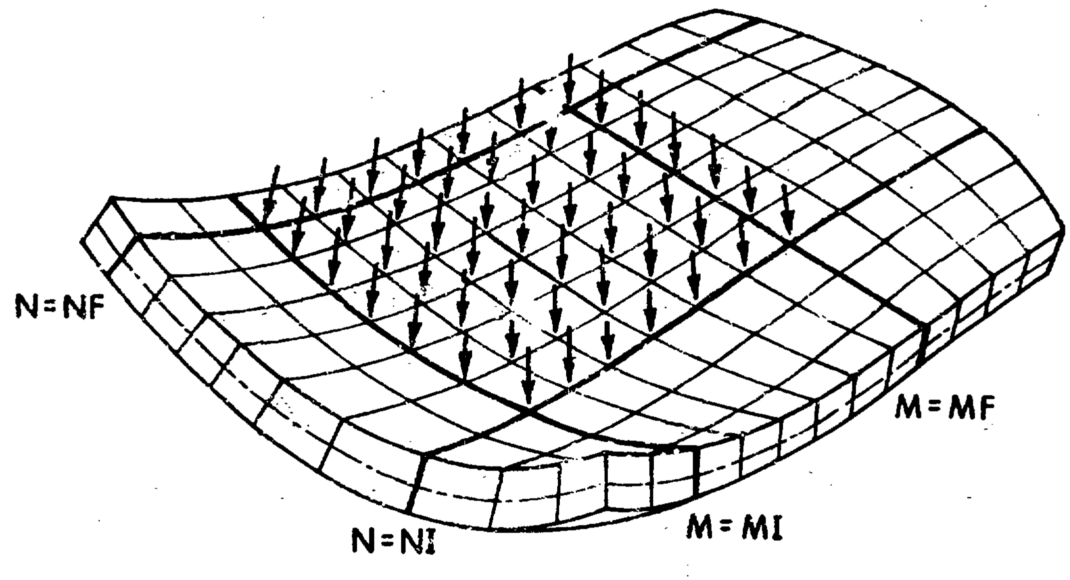


Figure 3.9 Uniform Initial Impulse Velocity Distributed over a Rectangular Region Defined by the Limits $MI \leqslant M \leqslant MF$ and $NI \leqslant N \leqslant NF$

VR — Uniform initial impulse velocity received by mesh points (M,N) in the intervals $MI \leq M \leq MF$ & $N1 \leq N \leq NF$. [L/T].

NV — Number of mesh point receiving nonzero initial impulse velocity other than uniform velocity VR.

Card 16* — Each card 16 gives data for each mesh point receiving nonuniform initial impulse velocity; hence total number of card 16's must equal NV.

M,N — Mesh numbers of point receiving initial impulse velocity V other than uniform velocity VR.

V — Initial impulse velocity at mesh point (M,N) [L/T].

3.3 Array Size, Memory Requirements and Computation Times

Often, the size of arrays must be adjusted in order for the program to accommodate a new problem. Such adjustments entail changing DIMENSION and COMMON statements to assure that array size equals or exceeds the maximum values of array indices required by the problem.

The maximum values of array indices are easily determined from the input data, as shall now be shown. First, denote the maximum values of the indices M,N,K,J and KJ (see Appendix C.1 for their definitions) by appending the subscript "max" to each. Then M_{max} and N_{max} depend on the number of mesh intervals in the η^1 and η^2 directions (specified on input card 2) and the boundary conditions along edges 3 and 2 respectively (card 4):

$$M_{max} = \begin{cases} MESH + 2 \; ; \; IBCE3 = 1,3 \\ MESH + 3 \; ; \; IBCE3 = 2 \end{cases} . \tag{3.1}$$

$$N_{max} = \begin{cases} NMESH + 1 \; ; \; IBCE2 = 1,3 \\ NMESH + 2 \; ; \; IBCE2 = 2 \end{cases} . \tag{3.2}$$

Moreover, K_{max} equals the number of layers into which the shell is divided (card 2):

$$K_{max} = LAYER \quad , \tag{3.3}$$

---

* Cards 15 and 16 omitted if LOAD = 1 or NCONT > 0.

J_{max} equals the number of breaks in the polygonal approximation to the loading curve (Card 6):

$$J_{max} = \text{NSFL}, \tag{3.4}$$

and KJ_{max} is the product of K_{max} and J_{max}:

$$KJ_{max} = \text{LAYER} * \text{NSFL} \quad . \tag{3.5}$$

The arrays affected by changes in the maximum values of these indices are listed in Appendices C.2 and C.4. The maximum values of the remaining array indices usually need no adjusting because they are sufficiently large for most problems.

The storage of arrays constitutes the major portion of the memory requirements of REPSIL, with the rest of the memory requirements used to store the remainder of the program. The memory needed to store the remainder is more or less fixed and depends on the compiler that the computer uses; for example, the BRLESC computer at the BRL uses between 11,000 and 12,000 words for this purpose. The memory required to store the arrays depends on their size, specified by prescribing the maximum values of array indicies, and hence may change with the problem being solved.

It is useful to have an estimate of the memory required by a problem, to see if the computer can accommodate it. For an estimate it is sufficient to consider only the large arrays — the two and three dimensional arrays. A count of these arrays, listed in Appendices C.2 and C.4, shows that there are 26 two dimensional (M,N) arrays, 3 three dimensional (M,N,KJ) arrays and 1 three dimensional (M,N,K) array. Hence, a problem whose maximum index values are M_{max}, N_{max}, K_{max} and $KJ_{max} \equiv K_{max} \times J_{max}$ will use

$$S_A = M_{max} \times N_{max} \times (26 + K_{max} \times (3 \times J_{max} + 1)) \tag{3.6}$$

words of memory to store these arrays. An estimate on the amount of memory a given computer uses to store the remainder of the program is most easily obtained as the difference between the total memory used on a given problem and S_A for the given problem. If this difference is S_R, then for any new problem a good estimate of the total number of words of memory needed is

$$S_T = S_R + S_A = S_R + M_{max} \times N_{max} \times (26 + K_{max} \times (3 \times J_{max} + 1)) . \tag{3.7}$$

The maximum values M_{max}, N_{max}, K_{max} and J_{max} can also be used to estimate the time a given problem will take on a computer. This follows from the fact that the computer takes approximately the same time to solve the finite difference scheme for a given substress at a given layer and mesh point; hence, the total computation time is roughly proportional to the product of the number of substresses J_{max}, of layers K_{max}, of mesh points $M_{max} \times N_{max}$ and the total number of time steps

$$N_t \equiv \text{MAXC} - \text{NCONT} . \tag{3.8}$$

Therefore, once the time T* taken by the computer to solve a problem with the values J^*_{max}, K^*_{max}, M^*_{max}, N^*_{max} and N^*_t is established, then an estimate of the time T for any other problem with the values J_{max}, K_{max}, M_{max}, N_{max} and N_t is given by the ratio

$$\frac{T}{T^*} = \frac{M_{max} \times N_{max} \times J_{max} \times K_{max} \times N_t}{M^*_{max} \times N^*_{max} \times J^*_{max} \times K^*_{max} \times N^*_t} . \tag{3.9}$$

This relation only gives approximate times because it neglects such factors as the operations at boundary point differing from those at interior points, the different number of interations for the plastic stress at different mesh points, the compile times not being proportional. However, the relation serves as a useful rule-of-thumb, giving over-estimates as the number of mesh intervals increases.

3.4 Continuation of Problem (Restart)

Every NRITE number of time steps information is written on the restart tape about the current state of the solution. As discussed in Chapter 2, this information is sufficient for the program to continue the solution from any of these prescribed time steps; such a continuation is called a restart. The restart of a problem entails certain necessary, advisable and permissible changes in the input data, as follow.

Necessary changes are changes without which the restart problem cannot be solved. These changes are confined to Card 3. The initial time step NCONT must be set equal to some multiple of NRITE at which time step there is written information on the restart tape; this is usually chosen as the last time step on the tape. Also, the last time step MAXC must be changed to a value greater than NCONT.

Advisable changes are changes that either reduce the amount of input data or assure meaningful output data. First, Cards 15 and 16 may be omitted. Second, the time steps specified on Cards 9, 10, and 11 at which output data is collected should be changed so as to fall within the new interval between NCONT and MAXC.

Permissible changes are changes that affect the collecting of restart and output data or the functioning of the damping operations. On Card 3 NRITE may be changed so that restart information is collected at a different interval. Also, on Card 8 the interval between the times at which surface strains are printed may be changed by altering NPRINT and a different selection of arrays may be printed by altering JCHK(J). On Card 5, DAMPF and DFACT, which control damping operations, may be freely changed. Also, if LPRESS and MDAMP were greater than the new value of NCONT, they may be changed.

Finally, a few words of caution. If it is planned to continue a problem, provide a tape to collect the restart information; otherwise, omit the restart tape. Any changes in the input data that affect the size of arrays should never be made. Lastly, note that there is no restriction on repeatedly continuing the solution of a problem.

4. DESCRIPTION OF OUTPUT

REPSIL outputs the results of calculations in two forms: printed output and plotted output. This chapter describes the various output options available and how these are controlled through the input data. Reference will be made to tables and figures in Chapter 5 as samples of printed and plotted output.

4.1 Printed Output

4.1.1 Input Data. The printed output begins with a title page reiterating the input data and giving the critical Δt resulting from the REPSIL stability check, see Tables 5.3 and 5.15. If LOAD = 0 or -1, the uniform initial impulse velocity specified on Cards 15 and 16 will be printed out next, Table 5.16. There follows a print of the initial values of the Cartesian coordinates Yi and pressure P at all mesh intersections (M,N). Table 5.4 and 5.16 give samples of these arrays at select values of M. The user need not request any of the above prints; they are automatically produced by REPSIL for any initial run. For a restart run only the title page is printed.

4.1.2 Displacement Increment, Cartesian Coordinate, Pressure and Surface Normal Arrays. By setting the input variables JCHK(I) = 1 for I = 1,2,3 (see Section 3.2, Card 8), the values of the displacement increments Ui, Cartesian coordinates Yi and pressures P, and surface normals SNi at every intersection (M,N) are printed at each NCYCH(J) time step (Card 9). Samples of these arrays at select values of M appear in Tables 5.5 - 5.7, 5.9 - 5.11 and 5.17 - 5.22. Notice that the displacement increments are for the time increment just preceding time step NCYCH(J).

4.1.3 Energy Balance. The program prints the values of the kinetic energy CINET, strain energy STREN, plastic work PLAST and external work TNRG for the entire shell every NCYCH(J) time steps (Section 3.2, Card 9), as shown in Tables 5.13 and 5.23.

4.1.4 Stress Subincrement Array. The stress subincrement array LMAT (M,N,K) is printed every JCYNLP(J) time steps (Section 3.2, Card 10), see Tables 5.8, 5.12, 5.18, 5.20 and 5.22. The value of LMAT at the location (M,N,K) is an approximate measure of the amount of plasticity occurring there during the given time interval: if LMAT = 0 the stress increment lies within the yield surface and, hence, is elastic; otherwise LMAT equal the number of stress annuli outside the yield surface traversed by the stress increment, see Appendix B for a detailed description.

4.1.5 Surface Strains. The surface strains EPSSI(I), EPSS2(I), EPSANB(I) and EPSANG(I) at locations specified on input Card 13 are printed every NPRINT elapsed time steps (Card 8). These strains, shown in Tables 5.13 and 5.23, simulate the reading of strain gages alined at the angles indicated there relative to the η^1 direction, see also Figure 3.5.

4.1.6 Error Messages. An inability to satisfactorally calculate a plastic stress increment at a location (M,N,K) results in an error print, wherein the values of the quantities involved in computing the stress at this location are printed and the calculations terminate. This print occurs for two reasons: either the lead coefficient AA in the quadratic equation for TAMBDA is negative or the values of TAMBDA continue to remain complex even after the use of 100 stress subdivision. The reasons that both these results are unacceptable and do not permit the continuation of the solution are given in [3; Sect III].

4.2 Plotted Output

The plotted output is generated by a separate plotting program described in Appendix D. This program works from a tape generated by REPSIL on which plotting data is stored. The program employs the Cal Comp Standard Plotting Package SCOOP. The plots shown in this report are generated by the Cal Comp Model 780 Plotter.

4.2.1 Isometric and Cross-sectional Plots. REPSIL stores on tape the Cartesian coordinate array Yi at the initial time step and at subsequent time steps as specified by the values of NC3DP(J) on Card 11. From these data the plotting program generates two types of plots at each of these time step: an isometric drawing of the distorted image of the finite difference mesh passing through the middle surface; and a pair of cross-sectional drawing through the Y1 = 0 symmetry plane and a plane normal to the Y2 axis, as specified in the input to the plotting program. The scale of the drawing and a factor to magnify the displacements from the initial position must also be specified as input. These plots are illustrated by Figures 5.3 and 5.8. The zeroth time step plots are

automatically generated without the user requesting them. Notice that these plots print the input data: the mesh number N of the crossection, the scale and the magnification factor.

4.2.2 Energy, Displacement and Surface Strain Histories. At every time step REPSIL also stores on tape three groups of data: the kinetic energy CINET, strain energy STREN, plastic work PLAST, external work TNRG and, when the damping procedure is used, the damping work TDAMP; the components of the displacement at the location specified on Card 12; and the elongational surface strains in the η^1 and η^2 coordinate direction at the location specified on Card 13.

The plotting program, using the first group of data, plots a time history of the balance of energies during deformation, as illustrated in Figures 5.4 and 5.9; notice that graph b of Figure 5.9 is an enlargement of the last 400 microseconds of graph a to bring out the details of the energy balance during damping operations. In Figures 5.4 and 5.9a the top line represents the external work, with that in Figure 5.4 being due to a pressure loading and in Figure 5.9a due to an initial impulse velocity; in Figure 5.9b the external work line falls of the graph. In all three graphs, the bottom line is the kinetic energy and the line second from the bottom is the total energy of the shell. Hence, the difference between the bottom line and the second from bottom line represents the strain energy and the difference between the top line and the second from bottom line represents the energy dissipated. When damping operations are not used, as in Figure 5.4 and the first 405 microseconds in Figure 5.9, the energy dissipated is solely due to the plastic work. However, at the inception of the damping procedure (405 microseconds in Figure 5.9b) a third line appears dividing the energy dissipated into two parts: the plastic work represented by the difference between the top line and this new line and the damping work represented by the remaining difference.*

The second group of data is used by the plotting program to plot a time history of the components of displacement at a given location, as illustrated in Figures 5.5 and 5.10. The location as specified in the REPSIL input is printed with the plot.

* In principle, the plastic work and the damping work are monotone increasing function of time and when the external work is constant the total energy is monotone decreasing. That this is only approximately true of Figure 5.9b is a consequence of the numerical inaccuracy of the finite difference solution. For the same reason, in purely elastic problems it is found that the total energy oscillates about the external work rather than coinciding. However, an excessive rise of the total energy over the external work is usually an indication that something is going wrong with the solution, e.g. too large a time increment leading to a numerical instability.

The third group is used to plot time histories of the elongational strains on the surface of the shell in the η^1 and η^2 coordinate directions (see Figures 3.6 - 3.8 for the coordinate directions of the various geometries employe, as well as Figure 3.5) at prescribed locations. These plots are illustrated by Figures 5.6 and 5.11. with the solid line being the strain in the η^1 direction and the dash line the strain in the η^2 direction. The locations as specified in REPSIL, including which bounding surface, are reproduced on the plots. A maximum of 6 locations can be plotted at present.

5. EXAMPLE PROBLEMS

This chapter demonstrates the use of REPSIL to solve two typical shell problems. Correct preparation of input data, including proper implementation of various options available in REPSIL is illustrated. Portions of the printed and plotted output are displayed. These serve as checks on the proper functioning of the code, especially useful to users adapting REPSIL to their computers. For reasons of economy, no attempt is made to discuss the accuracy or significance of results.

5.1 Example 1: Pressure Loaded Flat Plate

The first problem involves finding the deformation history of a simple-supported, rectangular, steel plate subjected to loads resulting from the detonation of an explosive charge. Figure 5.1 shows the dimensions and orientation of the plate and charge. This example illustrates the use of the following REPSIL options.

- Flat plate initial geometry
- Analytically specified pressure loading
- Hinged edge and symmetry edge boundaries
- Strain hardening - strain rate dependent material behavior
- Problem restart or continuation

The material properties of the steel in the elastic range are:

Young's modulus $E = 30 \times 10^6$ psi

Poisson's ratio $\nu = 0.3$

Mass density $\rho = 7.3235 \times 10^{-4} \dfrac{\text{lb - sec}^2}{\text{in}^4}$

In the plastic range the steel is assumed to strain harden in a strain rate dependent manner. This behavior is approximated by using a 3 substress model with the following values of the parameters.

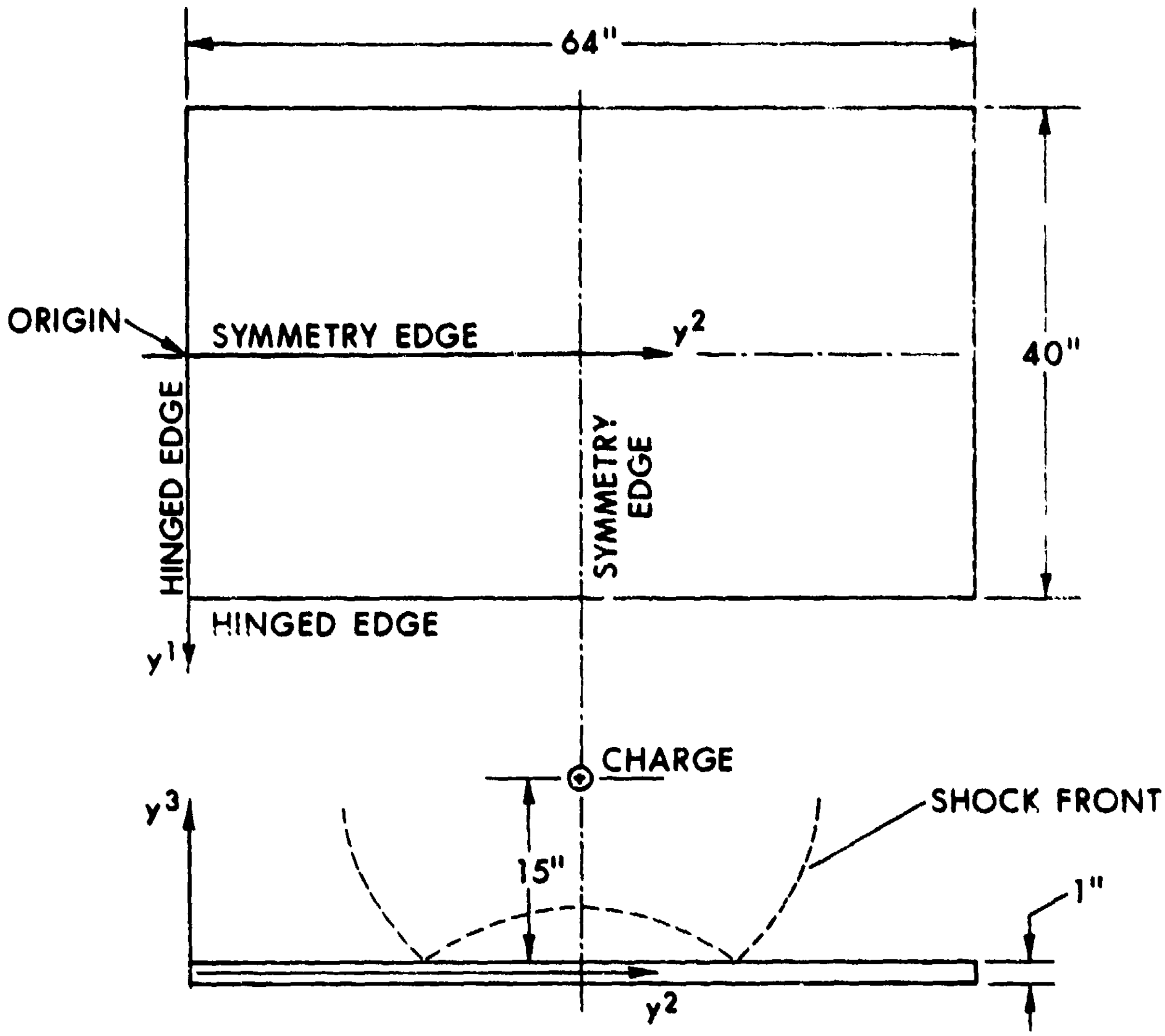


Figure 5.1 Geometry for Example Problem 1

σ_1 = 78,000 psi $\quad \varepsilon_1$ = .0026 in/in $\quad d_1$ = 40 in/in-sec $\quad P_1$=5

σ_2 = 120,000 psi $\quad \varepsilon_2$ = .0082 in/in $\quad d_2$ = 400 in/in-sec $\quad P_2$=3

σ_3 = 180,000 psi $\quad \varepsilon_3$ = .0482 in/in $\quad d_3$ = 4000 in/in-sec $\quad P_3$=1

Figure 5.2 shows the resulting polygonal approximations to the uniaxial loading curves at four strain rate levels.

The pressure loading is simulated by programming into subroutine PRESS the pressure relation

$$P\,(m,n) = \begin{cases} 0;\ \text{for } t < t_a \\ \dfrac{225 \times 24465.5}{225 + r^2\,(m,n)}\ e^{-13000(t-t_a)}\ ;\ \text{for } t \geq t_a, \end{cases}$$

where $r\,(m,n)$ is the distance from the center of the plate to the mesh intersection (m,n) and

$$t_a = \frac{\sqrt{225 + r^2\,(m,n)} - 15}{144000}$$

is the arrival time of the shock front at (m,n).

Taking advantage of the two fold symmetry of the problem, only the lower right quarter of the plate, as shown in Figure 5.1, is treated. Consequently, edges 1 and 2 are symmetry boundaries and edges 3 and 4 hinged boundaries (compare with Figures 3.1 and 3.6). Also, only the half width and half length of the plate are prescribed as input dimensions. The problem uses a 20 x 32 square mesh and 4 layers through the thickness. A time increment of 4 microseconds is prescribed, a figure well below the critical time increment predicted by the REPSIL stability criteria (see Table 5.3).

The problem is solved in three successive runs. The initial run is set for a maximum of 400 time steps, giving a solution for the first 1600 microseconds. Table 5.1 gives the input for this run in the same order as outlined in Table 3.1. The first restart or continuation run is prescribed for the next 400 time step. Only the input data on Cards 3, 9, 10 and 11 are changed, as shown in Table 5.2. The second restart run continues the solution 389 time steps further and requires changes in Card 3, 9, 10 and 11 as before.

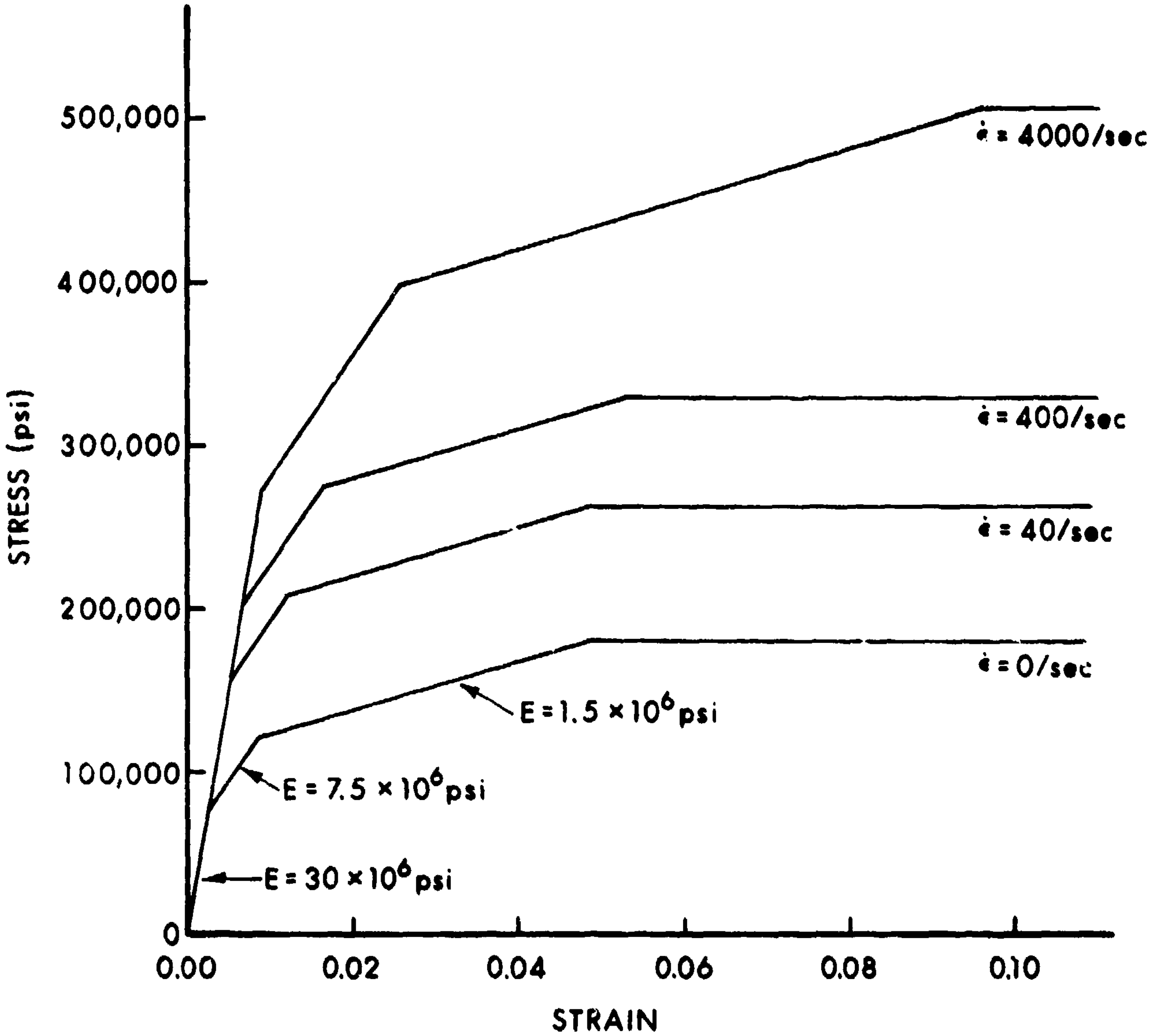


Figure 5.2 Polygonal Approximations to Uniaxial Loading Curves at Four Levels of Strain Rate

Table 5.1 Input Data Cards for Initial Run of Flat Plate Problem

Card 1	EXAMPLE 1 FLAT PLATE WITH PRESSURE LOADING							
Card 2	20 32 4 .100000E 01							
Card 3	400 0 100 .400000E-05							
Card 4	2 2 3 3							
Card 5	1 1500 1500 0.0	0.0						
Card 6	.300000E 08	.300000E 00	.780000E 05	.732350E-03	.100000E 00	3	1	
Card 7a	78000.0	.0026	40.0	5.0				
Card 7b	120000.0	.0082	400.0	3.0				
Card 7c	180000.0	.0482	4000.0	1.0				
Card 8	100	1	1	1				
Card 9	4	100	200	300	400			
Card 10	4	100	200	300	400			
Card 11	4	100	200	300	400			
Card 12	0.0	32.0	6					
Card 13a	0.0	32.0	45.0	22.5	0			
Card 13b	0.0	32.0	45.0	22.5	1			
Card 13c	0.0	0.0	45.0	22.5	0			
Card 13d	0.0	0.0	45.0	22.5	1			
Card 13e	20.0	32.0	45.0	22.5	0			
Card 13f	20.0	32.0	45.0	22.5	1			
Card 14	32.0	20.0						

Table 5.2 Input Data Cards Changed for the First Restart Run

Card 3	800	400	100	.400000E-05	
Card 9	4	500	600	700	800
Card 10	4	500	600	700	800
Card 11	4	500	600	700	800

Sample listings of the printed output for the initial run are given in Tables 5.3 - 5.13. The program prints the Ui (M,N), Yi (M,N), SNi (M,N) arrays over the entire range of (M,N), of which Tables 5.4 - 5.7 and 5.9 - 5.11 give the values for M = 2,10 and $1 \leq N \leq 33$ (i.e. the values along $y^1 = 0$ and $y^1 = 8$, see Figure 5.1). Also the program prints the LMAT (M,N,K) array at every station through the thickness, of which Tables 5.6 and 5.12 give the array values at K = 3,4. Table 5.13 shows the surface strains and energy balance prints.

Figures 5.3 - 5.6 present examples of the plotted output as obtained on the Cal Comp Model 780 Plotter using the plotting program described in Appendix E. Notice that the plots with time as the abscissa, Figures 5.4 - 5.6, are for the initial run plus the two successive restarts, an automatic feature of the plotting program.

Table 5.3 First Page of Printed Output Summarizing the Input Data and Results of Stable Time Increment Check

```
                    BRL REPSIL CODE

            EXAMPLE 1  FLAT PLATE WITH PRESSURE LOADING

      20 MESHES IN THE ETA1 DIRECTION    (DETA1 =0.100000E 01)
      32 MESHES IN THE ETA2 DIRECTION    (DETA2 =0.100000E 01)

              BENDING  TIME INCREMENT= 0.514304E-05
              MEMBRANE TIME INCREMENT= 0.666553E-05
              INPUT    TIME INCREMENT= 0.400000E-05

            TIME INCREMENT USED BY REPSIL= 0.400000E-05

                         YOUNG'S MODULUS =0.300000E 08
POISSON'S RATIO =0.300000E 00           YIELD STRESS     =0.780000E 05
MASS DENSITY    =0.732350E-03           THICKNESS        =0.100000E 01

                      START AT TIME STEP    0
                      FINAL  TIME  STEP   400
                      SURFACE STRAINS EVERY  100 TIME STEP
                      RESTART  WRITE  EVERY  100 TIME STEP

            LAYER =    4                  NSTRN  =    6
            LOAD  =    1                  LPRESS = 1500

                      BOUNDARY CONDITIONS
              1/2/3 = CLAMPED/SYMMETRY/HINGED
                         EDGE1 =    2
                         EDGE2 =    2
                         EDGE3 =    3
                         EDGE4 =    3

                  PRINT OPTION CONTROL CARD
                    0/1 = NO PRINT/PRINT
                     1 DISPLACEMENT INCREMENTS
                     1 CARTESIAN COORDINATES, PRESSURE
                     1 SURFACE NORMAL VECTOR COMPONENTS

PRINT INFORMATION AT THE FOLLOWING TIME STEPS
  100  200  300  400
PRINT L MATRIX (LMAT) AT THE FOLLOWING TIME STEPS
  100  200  300  400
3-D PLOTS FOR THE FOLLOWING TIME STEPS
  100  200  300  400
          CONSTITUTIVE RELATION    ELASTOPLASTIC-WORK HARDENING-STRAIN RATE DEPENDENT
        STRESS-STRAIN APPROXIMATION HAS  3 SUBLAYERS

                  STRESS-STRAIN AND STRAIN RATE PARAMETERS
     J          SSIG(J)        SEPS(J)              DSR(J)           1/PSR(J)
     1       7.8000000E 04  2.6000000E-03       4.0000000E 01   2.0000000E-01
     2       1.2000000E 05  8.2000000E-03       4.0000000E 02   3.3333333E-01
     3       1.8000000E 05  4.8200000E-02       4.0000000E 03   1.0000000E 00

                 START DAMPING AFTER TIME STEP 1500     TIME =0.6000E-02
          DAMPF =0.0000E 00                   DFACT =0.0000E 00
```

Table 5.4 Sample of Initial Values of Cartesian Coordinates and Pressure Arrays for M = 2 and 10

M	N	INITIAL CARTESIAN COORDINATES Y1(M,N)	Y2(M,N)	Y3(M,N)	PRESSURE P(M,N)
2	1	0.0000000000000000E 00	0.0000000000000000E 00	0.0000000000000000E 00	0.0000000000000000E 00
	2	0.0000000000000000E 00	0.1000000000000000E 01	0.0000000000000000E 00	0.0000000000000000E 00
	3	0.0000000000000000E 00	0.2000000000000000E 01	0.0000000000000000E 00	0.0000000000000000E 00
	4	0.0000000000000000E 00	0.3000000000000000E 01	0.0000000000000000E 00	0.0000000000000000E 00
	5	0.0000000000000000E 00	0.4000000000000000E 01	0.0000000000000000E 00	0.0000000000000000E 00
	6	0.0000000000000000E 00	0.5000000000000000E 01	0.0000000000000000E 00	0.0000000000000000E 00
	7	0.0000000000000000E 00	0.6000000000000000E 01	0.0000000000000000E 00	0.0000000000000000E 00
	8	0.0000000000000000E 00	0.7000000000000000E 01	0.0000000000000000E 00	0.0000000000000000E 00
	9	0.0000000000000000E 00	0.8000000000000000E 01	0.0000000000000000E 00	0.0000000000000000E 00
	10	0.0000000000000000E 00	0.9000000000000000E 01	0.0000000000000000E 00	0.0000000000000000E 00
	11	0.0000000000000000E 00	0.1000000000000000E 02	0.0000000000000000E 00	0.0000000000000000E 00
	12	0.0000000000000000E 00	0.1100000000000000E 02	0.0000000000000000E 00	0.0000000000000000E 00
	13	0.0000000000000000E 00	0.1200000000000000E 02	0.0000000000000000E 00	0.0000000000000000E 00
	14	0.0000000000000000E 00	0.1300000000000000E 02	0.0000000000000000E 00	0.0000000000000000E 00
	15	0.0000000000000000E 00	0.1400000000000000E 02	0.0000000000000000E 00	0.0000000000000000E 00
	16	0.0000000000000000E 00	0.1500000000000000E 02	0.0000000000000000E 00	0.0000000000000000E 00
	17	0.0000000000000000E 00	0.1600000000000000E 02	0.0000000000000000E 00	0.0000000000000000E 00
	18	0.0000000000000000E 00	0.1700000000000000E 02	0.0000000000000000E 00	0.0000000000000000E 00
	19	0.0000000000000000E 00	0.1800000000000000E 02	0.0000000000000000E 00	0.0000000000000000E 00
	20	0.0000000000000000E 00	0.1900000000000000E 02	0.0000000000000000E 00	0.0000000000000000E 00
	21	0.0000000000000000E 00	0.2000000000000000E 02	0.0000000000000000E 00	0.0000000000000000E 00
	22	0.0000000000000000E 00	0.2100000000000000E 02	0.0000000000000000E 00	0.0000000000000000E 00
	23	0.0000000000000000E 00	0.2200000000000000E 02	0.0000000000000000E 00	0.0000000000000000E 00
	24	0.0000000000000000E 00	0.2300000000000000E 02	0.0000000000000000E 00	0.0000000000000000E 00
	25	0.0000000000000000E 00	0.2400000000000000E 02	0.0000000000000000E 00	0.0000000000000000E 00
	26	0.0000000000000000E 00	0.2500000000000000E 02	0.0000000000000000E 00	0.0000000000000000E 00
	27	0.0000000000000000E 00	0.2600000000000000E 02	0.0000000000000000E 00	0.0000000000000000E 00
	28	0.0000000000000000E 00	0.2700000000000000E 02	0.0000000000000000E 00	0.0000000000000000E 00
	29	0.0000000000000000E 00	0.2800000000000000E 02	0.0000000000000000E 00	0.0000000000000000E 00
	30	0.0000000000000000E 00	0.2900000000000000E 02	0.0000000000000000E 00	0.0000000000000000E 00
	31	0.0000000000000000E 00	0.3000000000000000E 02	0.0000000000000000E 00	0.0000000000000000E 00
	32	0.0000000000000000E 00	0.3100000000000000E 02	0.0000000000000000E 00	0.0000000000000000E 00
	33	0.0000000000000000E 00	0.3200000000000000E 02	0.0000000000000000E 00	0.2446550000000000E 05
10	1	0.8000000000000000E 01	0.0000000000000000E 00	0.0000000000000000E 00	0.0000000000000000E 00
	2	0.8000000000000000E 01	0.1000000000000000E 01	0.0000000000000000E 00	0.0000000000000000E 00
	3	0.8000000000000000E 01	0.2000000000000000E 01	0.0000000000000000E 00	0.0000000000000000E 00
	4	0.8000000000000000E 01	0.3000000000000000E 01	0.0000000000000000E 00	0.0000000000000000E 00
	5	0.8000000000000000E 01	0.4000000000000000E 01	0.0000000000000000E 00	0.0000000000000000E 00
	6	0.8000000000000000E 01	0.5000000000000000E 01	0.0000000000000000E 00	0.0000000000000000E 00
	7	0.8000000000000000E 01	0.6000000000000000E 01	0.0000000000000000E 00	0.0000000000000000E 00
	8	0.8000000000000000E 01	0.7000000000000000E 01	0.0000000000000000E 00	0.0000000000000000E 00
	9	0.8000000000000000E 01	0.8000000000000000E 01	0.0000000000000000E 00	0.0000000000000000E 00
	10	0.8000000000000000E 01	0.9000000000000000E 01	0.0000000000000000E 00	0.0000000000000000E 00
	11	0.8000000000000000E 01	0.1000000000000000E 02	0.0000000000000000E 00	0.0000000000000000E 00
	12	0.8000000000000000E 01	0.1100000000000000E 02	0.0000000000000000E 00	0.0000000000000000E 00
	13	0.8000000000000000E 01	0.1200000000000000E 02	0.0000000000000000E 00	0.0000000000000000E 00
	14	0.8000000000000000E 01	0.1300000000000000E 02	0.0000000000000000E 00	0.0000000000000000E 00
	15	0.8000000000000000E 01	0.1400000000000000E 02	0.0000000000000000E 00	0.0000000000000000E 00
	16	0.8000000000000000E 01	0.1500000000000000E 02	0.0000000000000000E 00	0.0000000000000000E 00
	17	0.8000000000000000E 01	0.1600000000000000E 02	0.0000000000000000E 00	0.0000000000000000E 00
	18	0.8000000000000000E 01	0.1700000000000000E 02	0.0000000000000000E 00	0.0000000000000000E 00
	19	0.8000000000000000E 01	0.1800000000000000E 02	0.0000000000000000E 00	0.0000000000000000E 00
	20	0.8000000000000000E 01	0.1900000000000000E 02	0.0000000000000000E 00	0.0000000000000000E 00
	21	0.8000000000000000E 01	0.2000000000000000E 02	0.0000000000000000E 00	0.0000000000000000E 00
	22	0.8000000000000000E 01	0.2100000000000000E 02	0.0000000000000000E 00	0.0000000000000000E 00
	23	0.8000000000000000E 01	0.2200000000000000E 02	0.0000000000000000E 00	0.0000000000000000E 00
	24	0.8000000000000000E 01	0.2300000000000000E 02	0.0000000000000000E 00	0.0000000000000000E 00
	25	0.8000000000000000E 01	0.2400000000000000E 02	0.0000000000000000E 00	0.0000000000000000E 00
	26	0.8000000000000000E 01	0.2500000000000000E 02	0.0000000000000000E 00	0.0000000000000000E 00
	27	0.8000000000000000E 01	0.2600000000000000E 02	0.0000000000000000E 00	0.0000000000000000E 00
	28	0.8000000000000000E 01	0.2700000000000000E 02	0.0000000000000000E 00	0.0000000000000000E 00
	29	0.8000000000000000E 01	0.2800000000000000E 02	0.0000000000000000E 00	0.0000000000000000E 00
	30	0.8000000000000000E 01	0.2900000000000000E 02	0.0000000000000000E 00	0.0000000000000000E 00
	31	0.8000000000000000E 01	0.3000000000000000E 02	0.0000000000000000E 00	0.0000000000000000E 00
	32	0.8000000000000000E 01	0.3100000000000000E 02	0.0000000000000000E 00	0.0000000000000000E 00
	33	0.8000000000000000E 01	0.3200000000000000E 02	0.0000000000000000E 00	0.0000000000000000E 00

Table 5.5 Sample of Displacement Increments Array During Time Increment 199 to 200 for M = 2 and 10

DISPLACEMENT INCREMENTS BETWEEN T.S. 199 AND 200

M	N	U1(M,N)	U2(M,N)	U3(M,N)
2	1	0.0000000000000000E 00	0.0000000000000000E 00	0.0000000000000000E 00
	2	0.0000000000000000E 00	-0.1381206393653739E-04	-0.4168488328986917E-03
	3	0.0000000000000000E 00	-0.2375343470663745E-04	-0.8474448913925587E-03
	4	0.0000000000000000E 00	-0.4720040488698901E-04	-0.1423860541853202E-02
	5	0.0000000000000000E 00	-0.5008229783695276E-04	-0.1779364250648565E-02
	6	0.0000000000000000E 00	-0.9719565487315917E-04	-0.2303383105088261E-02
	7	0.0000000000000000E 00	-0.8292596569873348E-04	-0.2736982933096866E-02
	8	0.0000000000000000E 00	-0.1024470004462928E-03	-0.3295136604822684E-02
	9	0.0000000000000000E 00	-0.1083337421897568E-03	-0.3708735256807441E-02
	10	0.0000000000000000E 00	-0.1310749969676948E-03	-0.4504531168145721E-02
	11	0.0000000000000000E 00	-0.1536371012469078E-03	-0.5020667473699230E-02
	12	0.0000000000000000E 00	-0.2164696823739369E-03	-0.5522972854020365E-02
	13	0.0000000000000000E 00	-0.2042007798745957E-03	-0.6104938562385423E-02
	14	0.0000000000000000E 00	-0.1892760728979844E-03	-0.6431512541719008E-02
	15	0.0000000000000000E 00	-0.2163818744548673E-03	-0.6974581950601418E-02
	16	0.0000000000000000E 00	-0.1545698046836804E-03	-0.7231816325527253E-02
	17	0.0000000000000000E 00	-0.2130450507842500E-03	-0.7705063805635238E-02
	18	0.0000000000000000E 00	-0.2293594139886983E-03	-0.7850570817561857E-02
	19	0.0000000000000000E 00	-0.1977772677079695E-03	-0.8172739998317206E-02
	20	0.0000000000000000E 00	-0.2003865942925829E-03	-0.8185992246671717E-02
	21	0.0000000000000000E 00	-0.1976957044141982E-03	-0.8011574519626505E-02
	22	0.0000000000000000E 00	-0.1176605779205775E-03	-0.7838382354107286E-02
	23	0.0000000000000000E 00	-0.2183472079786409E-03	-0.7625775726690378E-02
	24	0.0000000000000000E 00	-0.1198983869082797E-03	-0.7495273269677910E-02
	25	0.0000000000000000E 00	-0.8685676002769613E-04	-0.7320251921932451E-02
	26	0.0000000000000000E 00	-0.1141018167286585E-03	-0.7177705436977719E-02
	27	0.0000000000000000E 00	-0.7751586966518130E-04	-0.7359178246913641E-02
	28	0.0000000000000000E 00	-0.5338714258114968E-04	-0.7323212650284600E-02
	29	0.0000000000000000E 00	-0.8783717403050155E-04	-0.7419899805486476E-02
	30	0.0000000000000000E 00	-0.7163153237329932E-04	-0.7551315721[illegible]22282E-02
	31	0.0000000000000000E 00	-0.1932272522140175E-04	-0.7910157657475296E-02
	32	0.0000000000000000E 00	0.6771410273254223E-04	-0.7656193690124109E-02
	33	0.0000000000000000E 00	0.0000000000000000E 00	-0.8156761964262543E-02
10	1	0.0000000000000000E 00	0.0000000000000000E 00	0.0000000000000000E 00
	2	-0.9967931064029214E-06	-0.1934307557308220E-04	-0.2212602323124786E-03
	3	-0.5019807073676166E-05	-0.8790685298558954E-05	-0.4354805558474219E-03
	4	-0.1463650466912884E-05	-0.1839791110262883E-04	-0.5046060999896601E-03
	5	0.2577635059557098E-04	0.5542688208307563E-05	-0.6259591087167490E-03
	6	-0.7932696708386931E-05	0.2139139744171304E-04	-0.8831368253513934E-03
	7	0.9736339864651590E-05	0.1013419980963647E-05	-0.1197763245342692E-02
	8	0.1444166746512143E-04	0.1895683078942489E-04	-0.1504622949863701E-02
	9	0.4252064687469055E-05	0.4529530138226414E-05	-0.1812568261607337E-02
	10	0.3797795316647549E-04	-0.1243705795220926E-04	-0.2218658947899959E-02
	11	0.2065600270568753E-04	0.1046976067768080E-04	-0.2479629271162489E-02
	12	0.3675218863317805E-04	-0.3867953625348610E-04	-0.2816299092642461E-02
	13	-0.1268702946147533E-05	-0.2245857195802538E-04	-0.3038635553045534E-02
	14	0.4087752628936709E-04	-0.5080187558670575E-04	-0.3462661713695602E-02
	15	0.5603815836442154E-04	-0.5771193611169779E-04	-0.3769302093188341E-02
	16	0.6593726844771896E-04	-0.4061778392411700E-04	-0.4133857448863767E-02
	17	0.1067078279420197E-03	-0.9097365160805708E-04	-0.4594870595072928E-02
	18	0.9490999490945175E-04	-0.9357838581299170E-04	-0.4910451120878080E-02
	19	0.1013614742883946E-03	-0.9812288277081476E-04	-0.5124806084896764E-02
	20	0.9890476997012357E-04	-0.6369638946751248E-04	-0.5326621093544308E-02
	21	0.1418853757259157E-03	-0.5487998986584590E-04	-0.5229782576825541E-02
	22	0.1754142348527070E-03	-0.2434882159727478E-04	-0.5173626422641814E-02
	23	0.1694491410846638E-03	-0.6753179360222226E-04	-0.4968211830931578E-02
	24	0.1404496526327821E-03	-0.8052488359613026E-04	-0.4991519678772019E-02
	25	0.1562774257328265E-03	-0.4054505851603422E-04	-0.4914070665289152E-02
	26	0.1019[illegible]6512511E-03	0.1517890049126742E-04	-0.4762491235036106E-02
	27	0.1560721740602215E-03	-0.4144627974546997E-04	-0.4788485323506877E-02
	28	0.1627043896126920E-03	-0.2098060196229650E-04	-0.4702140416604566E-02
	29	0.2069522821120799E-03	-0.1194682230405943E-04	-0.4716081576773645E-02
	30	0.2219073771893363E-03	-0.1061741629519551E-03	-0.4879693121440284E-02
	31	0.2211373807693563E-03	0.5802377953685049E-04	-0.4767498558516651E-02
	32	0.1565500647510459E-03	0.3517116302069396E-04	-0.4659847749990656E-02
	33	0.3125816182922287E-03	0.0000000000000000E 00	-0.4805206926758087E-02

Table 5.6 Sample of Cartesian Coordinates and Pressure Arrays at Time Step 200 for M = 2 and 10

TIME STEP 200 TIME 0.80000000E-03

M	N	CARTESIAN COORDINATES V1(M,N, 200)	V2(M,N, 200)	V3(M,N, 200)	PRESSURE P(M,N)
2	1	0.0000000000000000E 00	0.0000000000000000E 00	0.0000000000000000E 00	0.0000000000000000E 00
	2	0.0000000000000000E 00	0.9999659602400206E 00	-0.5787704347785275E-01	0.8167919806216518E 00
	3	0.0000000000000000E 00	0.1999923771346841E 01	-0.1154194180527340E 00	0.7940755916351119E 00
	4	0.0000000000000000E 00	0.2999951592891313E 01	-0.1735278702788973E 00	0.7732311151081857E 00
	5	0.0000000000000000E 00	0.3999944225224123E 01	-0.2321204121180228E 00	0.7541906676816129E 00
	6	0.0000000000000000E 00	0.4999819464485916E 01	-0.2908728205803803E 00	0.736894264[illegible]8162E 00
	7	0.0000000000000000E 00	0.5999734150757253E 01	-0.3497134110940949E 00	0.7212894716344779E 00
	8	0.0000000000000000E 00	0.6999629057411831E 01	-0.4088972533546703E 00	0.7073310502354542E 00
	9	0.0000000000000000E 00	0.7999518618175310E 01	-0.4680544849847771E 00	0.6949805537267748E 00
	10	0.0000000000000000E 00	0.8999332266246000E 01	-0.5274922347870467E 00	0.6842058641830763E 00
	11	0.0000000000000000E 00	0.9999124364245030E 01	-0.5872477041867224E 00	0.6749806236460235E 00
	12	0.0000000000000000E 00	0.10998733584[illegible]44E 02	-0.6469786637248207E 00	0.6672835121779712E 00
	13	0.0000000000000000E 00	0.1199857482334034E 02	-0.7058701201216622E 00	0.6610973131379158E 00
	14	0.0000000000000000E 00	0.1299821908145814E 02	-0.7634156479555029E 00	0.6564076886322452E 00
	15	0.0000000000000000E 00	0.1399810552355361E 02	-0.8203182542318846E 00	0.6532015720448323E 00
	16	0.0000000000000000E 00	0.1499790077229687E 02	-0.8762680872269327E 00	0.6514650674768556E 00
	17	0.0000000000000000E 00	0.1599768831595512E 02	-0.9313939286794805E 00	0.6511807318168086E 00
	18	0.0000000000000000E 00	0.1699747641950878E 02	-0.9854320513624752E 00	0.6523241091540835E 00
	19	0.0000000000000000E 00	0.1799732157523514E 02	-0.1039145206000769E 01	0.6548593978182595E 00
	20	0.0000000000000000E 00	0.1899702545286974E 02	-0.1091934426954038E 01	0.6587341700359109E 00
	21	0.0000000000000000E 00	0.1999693108628917E 02	-0.1143305519578919E 01	0.6638731501693902E 00
	22	0.0000000000000000E 00	0.2099675662675275E 02	-0.1193162672642931E 01	0.6701712100147296E 00
	23	0.0000000000000000E 00	0.2199646436769383E 02	-0.1241584269589963E 01	0.6774859788583391E 00
	24	0.0000000000000000E 00	0.2299653673669818E 02	-0.1288421227415664E 01	0.6856308008228533E 00
	25	0.0000000000000000E 00	0.2399629462953092E 02	-0.1333424987944081E 01	0.6943691874873650E 00
	26	0.0000000000000000E 00	0.2499631710599600E 02	-0.1376328957298834E 01	0.7034123458249726E 00
	27	0.0000000000000000E 00	0.2599630172617388E 02	-0.1416967904244330E 01	0.7124216800305318E 00
	28	0.0000000000000000E 00	0.2699655985285373E 02	-0.1454564612176837E 01	0.7210181691608256E 00
	29	0.0000000000000000E 00	0.2799682030232250E 02	-0.1487699217488606E 01	0.7287999776766728E 00
	30	0.0000000000000000E 00	0.2899739719605393E 02	-0.1515616466768002E 01	0.7353683952219711E 00
	31	0.0000000000000000E 00	0.2999807554187096E 02	-0.1536646919434049E 01	0.7403602765472827E 00
	32	0.0000000000000000E 00	0.3099913507074701E 02	-0.1549453092611953E 01	0.7434829591273441E 00
	33	0.0000000000000000E 00	0.3200000000000000E 02	-0.1554093976030353E 01	0.7445459130421148E 00
10	1	0.8000000000000000E 01	0.0000000000000000E 00	0.0000000000000000E 00	0.0000000000000000E 00
	2	0.7999732288727414E 01	0.9999373396170459E 00	-0.5149942277337489E-01	0.8418575908690388E 00
	3	0.7999590794683458E 01	0.1999912262528011E 01	-0.1031642973483420E 00	0.8179390275934616E 00
	4	0.7999536726516393E 01	0.2999829982678408E 01	-0.1552214871938103E 00	0.7958936023236643E 00
	5	0.7999790304261775E 01	0.3999814784672533E 01	-0.2072499291925028E 00	0.7756447926294503E 00
	6	0.8000086257355872E 01	0.4999748951929792E 01	-0.2592449504297181E 00	0.7571231874321819E 00
	7	0.8000492759399781E 01	0.5999702780932307E 01	-0.3110199833167050E 00	0.7402660064382205E 00
	8	0.8001015429169993E 01	0.6999647236652598E 01	-0.3628448816667178E 00	0.7250165950967601E 00
	9	0.8001667152341545E 01	0.7999536035512374E 01	-0.4150536777016116E 00	0.7113238764962778E 00
	10	0.8002431531970063E 01	0.8999357261968186E 01	-0.4678280098973082E 00	0.6991417376968479E 00
	11	0.8003305992049259E 01	0.9999241616798461E 01	-0.5205929355063548E 00	0.6884283237080068E 00
	12	0.8004221716739729E 01	0.1099895057387506E 02	-0.5726151309326850E 00	0.6791452070869633E 00
	13	0.8005180395642457E 01	0.1199892013007349E 02	-0.6232931835693734E 00	0.6712563965338985E 00
	14	0.8006250598936349E 01	0.1299864029760508E 02	-0.6729463838276628E 00	0.6647271429937266E 00
	15	0.8007273446325064E 01	0.1399861653185095E 02	-0.7216236835841472E 00	0.6595224989188151E 00
	16	0.8008465342822085E 01	0.1499840784600662E 02	-0.7696297739780766E 00	0.6556055863873217E 00
	17	0.8009655283412928E 01	0.1599835272143833E 02	-0.8166219981386795E 00	0.6529355353375289E 00
	18	0.8010871304955921E 01	0.1699813101678216E 02	-0.8627041446649915E 00	0.6514650674768556E 00
	19	0.8012060269510901E 01	0.1799805803583359E 02	-0.9078296898116915E 00	0.6511377286251281E 00
	20	0.8013294901527412E 01	0.1899783435451526E 02	-0.9518733515048169E 00	0.6518848172342964E 00
	21	0.8014521506089777E 01	0.1999784269054536E 02	-0.9942925143136804E 00	0.6536221241927485E 00
	22	0.8015301097612963E 01	0.2099767461723362E 02	-0.1035153785756821E 01	0.6562466918011299E 00
	23	0.8017348073530956E 01	0.2199764641700348E 02	-0.1074382423047936E 01	0.6596339164048673E 00
	24	0.8018707116027690E 01	0.2299743325580055E 02	-0.1112274072629487E 01	0.6636354499472125E 00
	25	0.8020208553114021E 01	0.2399751727890038E 02	-0.1148709762267205E 01	0.6680784786228699E 00
	26	0.8021590198948941E 01	0.2499745502537334E 02	-0.1183757954084259E 01	0.6727670346418082E 00
	27	0.8023205770228908E 01	0.2599759027155899E 02	-0.1217172482523426E 01	0.6774859788583391E 00
	28	0.8024477186090069E 01	0.2699770094731253E 02	-0.1248121777988287E 01	0.6820081223591669E 00
	29	0.8025939743951624E 01	0.2799788291673956E 02	-0.1275942349123790E 01	0.6861045949049458E 00
	30	0.8026915356775315E 01	0.2899823638221196E 02	-0.1299234403183508E 01	0.6895580202247872E 00
	31	0.8027934470152458E 01	0.2999879817848022E 02	-0.1316989443910241E 01	0.6921774026142735E 00
	32	0.8028359956594166E 01	0.3099931135186901E 02	-0.1328353277272808E 01	0.6938130129742256E 00
	33	0.8028805322487589E 01	0.3200000000000000E 02	-0.1332175654565791E 01	0.6943691874873650E 00

Table 5.7 Sample of Surface Normal Components Array at Time Step 200 for M = 2 and 10

TIME STEP 200 TIME 0.8000000000E-03

SURFACE NORMAL VECTOR COMPONENTS

M	N	SN1(M,N)	SN2(M,N)	SN3(M,N)
2	1	0.0000000000000000E 00	0.5794857481663093E-01	0.9983195694148850E 00
	2	0.0000000000000000E 00	0.5761603559888880E-01	0.9983388162824138E 00
	3	0.0000000000000000E 00	0.5772939082038021E-01	0.9983322680528301E 00
	4	0.0000000000000000E 00	0.5825372372086324E-01	0.9983018099115385E 00
	5	0.0000000000000000E 00	0.5857560281164920E-01	0.9982829752906997E 00
	6	0.0000000000000000E 00	0.5869835111796001E-01	0.9982757652953579E 00
	7	0.0000000000000000E 00	0.5891532016478697E-01	0.9982629839124961E 00
	8	0.0000000000000000E 00	0.5907356954013085E-01	0.9982536317899309E 00
	9	0.0000000000000000E 00	0.5920226871409443E-01	0.9982460074446100E 00
	10	0.0000000000000000E 00	0.5950259421219625E-01	0.9982281509164218E 00
	11	0.0000000000000000E 00	0.5965477281262225E-01	0.9982190681812656E 00
	12	0.0000000000000000E 00	0.5922352270819250E-01	0.9982447467219811E 00
	13	0.0000000000000000E 00	0.5813488481200881E-01	0.9983087373993550E 00
	14	0.0000000000000000E 00	0.5714396880024767E-01	0.9983659483524849E 00
	15	0.0000000000000000E 00	0.5634554452449292E-01	0.9984113278665453E 00
	16	0.0000000000000000E 00	0.5546391738165598E-01	0.9984606922000890E 00
	17	0.0000000000000000E 00	0.5451239004557585E-01	0.9985130942213623E 00
	18	0.0000000000000000E 00	0.5380745730216241E-01	0.9985513294461513E 00
	19	0.0000000000000000E 00	0.5318780513532761E-01	0.9985845269104094E 00
	20	0.0000000000000000E 00	0.5201994904933800E-01	0.9986460458545381E 00
	21	0.0000000000000000E 00	0.5055619382739656E-01	0.9987212179911304E 00
	22	0.0000000000000000E 00	0.4909143323011414E-01	0.9987942887218635E 00
	23	0.0000000000000000E 00	0.4758056388384516E-01	0.9988674035829255E 00
	24	0.0000000000000000E 00	0.4587590520799637E-01	0.9989471464103327E 00
	25	0.0000000000000000E 00	0.4391628158072367E-01	0.9990352147007244E 00
	26	0.0000000000000000E 00	0.4173491537729273E-01	0.9991287188538073E 00
	27	0.0000000000000000E 00	0.3908319619295600E-01	0.9992359600091178E 00
	28	0.0000000000000000E 00	0.3533441056067107E-01	0.9993755447429010E 00
	29	0.0000000000000000E 00	0.3049895754374618E-01	0.9995347985881957E 00
	30	0.0000000000000000E 00	0.2445118785445006E-01	0.9997010250132319E 00
	31	0.0000000000000000E 00	0.1690121029083301E-01	0.9998571643443403E 00
	32	0.0000000000000000E 00	0.8714811380069492E-02	0.9999620253102665E 00
	33	0.0000000000000000E 00	0.0000000000000000E 00	0.1000000000000000E 01
10	1	0.0000000000000000E 00	0.5135303855177731E-01	0.9986805622577925E 00
	2	-0.1669875511733491E-02	0.5151550387932586E-01	0.9986707988000022E 00
	3	-0.3433129301139298E-02	0.5179373465996648E-01	0.9986519026530844E 00
	4	-0.5315371846915252E-02	0.5197480191562081E-01	0.9986342507584856E 00
	5	-0.7068453344170014E-02	0.5194390815117043E-01	0.9986249883562444E 00
	6	-0.8883560520375210E-02	0.5181929040065411E-01	0.9986169653550119E 00
	7	-0.1084420144573609E-01	0.5173521629106604E-01	0.9986019580845619E 00
	8	-0.1272981648847809E-01	0.5195418283068255E-01	0.9985683324933582E 00
	9	-0.1482421940176093E-01	0.5243168492688684E-01	0.9985144770782527E 00
	10	-0.1678118036782868E-01	0.5271034011176938E-01	0.9984688337803860E 00
	11	-0.1892192880714826E-01	0.5233993274507905E-01	0.9984500448447374E 00
	12	-0.2092210665899830E-01	0.5129911142981254E-01	0.998464157[illegible]81061E 00
	13	-0.2308955639646668E-01	0.5012038509115521E-01	0.9984762487829786E 00
	14	-0.2523982722416899E-01	0.4912408246675829E-01	0.9984737230610994E 00
	15	-0.2740011884383273E-01	0.4830305198158600E-01	0.9984568336471078E 00
	16	-0.2967589225060644E-01	0.4746629210724347E-01	0.9984319168239367E 00
	17	-0.3171868369336157E-01	0.4650798143124404E-01	0.9984142090174535E 00
	18	-0.3395643679461171E-01	0.4557765871688824E-01	0.9983835121966411E 00
	19	-0.3615365871932697E-01	0.4456155915458730E-01	0.9983522326348144E 00
	20	-0.3840959818430712E-01	0.4321293880032829E-01	0.9983272733365341E 00
	21	-0.4075611307752912E-01	0.4162796652239704E-01	0.9983015835157338E 00
	22	-0.4320748193769214E-01	0.4093825031897457E-01	0.9982635183236933E 00
	23	-0.4536163035890641E-01	0.3855777476909315E-01	0.9982262371306604E 00
	24	-0.4759602745287188E-01	0.3716631855114924E-01	0.9981749761119338E 00
	25	-0.4986118137038644E-01	0.3576105269169298E-01	0.9981157296139218E 00
	26	-0.5215152728775184E-01	0.3424158140153603E-01	0.9980519687394423E 00
	27	-0.5400547021522983E-01	0.3217728766428886E-01	0.9980220595484860E 00
	28	-0.5563692932312091E-01	0.2939845058623659E-01	0.9980181677698171E 00
	29	-0.5699550374619865E-01	0.2556903728477747E-01	0.9980469612623682E 00
	30	-0.5798106291915533E-01	0.2053311650318789E-01	0.9981065010656085E 00
	31	-0.5902355804941417E-01	0.1456730969411512E-01	0.9981502958389898E 00
	32	-0.5950213295317994E-01	0.7600774562291877E-02	0.9981992408392105E 00
	33	-0.5988980305902625E-01	0.0000000000000000E 00	0.9982050547197992E 00

Table 5.8 **Sample of Stress Subincrement Array at Time Step 200 for Layers K = 3 and 4**

LMAT(M,N, 3)

N	M= 2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
12	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
13	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
14	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
17	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
18	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
19	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
20	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
21	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
22	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
23	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
24	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
25	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
26	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
27	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0
28	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
29	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
30	1	0	0	0	1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
31	0	1	1	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
32	0	1	0	0	1	0	0	1	0	0	0	0	1	1	0	0	0	0	0	0	0
33	0	1	0	0	1	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0

LMAT(M,N, 4)

N	M= 2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
12	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
13	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
14	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
17	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
18	0	0	0	0	0	0	0	0	0	1	0	0	1	0	0	0	0	0	0	0	0
19	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
20	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0	0	0	0	0	0	0
21	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0	0	0	0
22	1	0	0	0	0	0	0	0	1	1	0	0	0	1	0	0	0	0	0	0	0
23	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
24	1	0	0	0	0	1	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0
25	1	0	0	1	0	0	0	0	0	1	0	0	0	0	1	0	0	0	0	0	0
26	1	0	0	1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
27	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
28	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
29	1	0	0	1	0	0	0	0	0	0	0	0	0	1	1	0	1	0	0	0	0
30	1	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
31	0	1	1	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0
32	0	1	0	0	1	0	0	1	0	0	0	0	1	0	1	1	0	0	0	0	0
33	0	1	0	0	1	0	0	0	0	0	0	1	0	0	1	0	1	0	0	0	0

Table 5.9 Sample of Displacement Increments Array During Time Increment 399 to 400 for M = 2 and 10

DISPLACEMENT INCREMENTS BETWEEN T.S. 399 AND 400

M	N	U1(M,N)	U2(M,N)	U3(M,N)
2	1	0.0000000000000000E 00	0.0000000000000000E 00	0.0000000000000000E 00
	2	0.0000000000000000E 00	-0.782708642608424E-04	-0.188741564554234E-03
	3	0.0000000000000000E 00	-0.524266336954022E-04	-0.152594689213415E-03
	4	0.0000000000000000E 00	0.125929290500402E-04	-0.335181150231113E-03
	5	0.0000000000000000E 00	0.687659937027248E-05	-0.192905342804471E-03
	6	0.0000000000000000E 00	-0.161252656134788E-03	-0.239799586538786E-03
	7	0.0000000000000000E 00	0.158338468116332E-04	-0.256665518061098E-03
	8	0.0000000000000000E 00	-0.135829914328585E-03	-0.374071564769151E-03
	9	0.0000000000000000E 00	-0.786396670423948E-04	-0.344013087214116E-03
	10	0.0000000000000000E 00	-0.983109434856791E-04	-0.451898191386336E-03
	11	0.0000000000000000E 00	-0.215638977365230E-03	-0.126240522148534E-03
	12	0.0000000000000000E 00	-0.200873890409324E-03	-0.951202533634292E-04
	13	0.0000000000000000E 00	-0.474745783755760E-04	0.301308468594181E-03
	14	0.0000000000000000E 00	0.812327112330738E-04	0.562297194517398E-03
	15	0.0000000000000000E 00	-0.196229186990204E-04	0.967681832527808E-03
	16	0.0000000000000000E 00	0.255105264971028E-04	0.125879213917149E-02
	17	0.0000000000000000E 00	0.516839072367544E-05	0.170830824542952E-02
	18	0.0000000000000000E 00	-0.668875509278022E-04	0.206287236140504E-02
	19	0.0000000000000000E 00	0.627118555549753E-04	0.250335748710145E-02
	20	0.0000000000000000E 00	-0.110767008761024E-03	0.291566804636881E-02
	21	0.0000000000000000E 00	0.742396971896478E-04	0.306265966376964E-02
	22	0.0000000000000000E 00	0.764398435449933E-04	0.329925707148542E-02
	23	0.0000000000000000E 00	-0.522568252751086E-04	0.331965585622838E-02
	24	0.0000000000000000E 00	-0.271979444651827E-04	0.326971645626319E-02
	25	0.0000000000000000E 00	0.520355927945684E-04	0.315202226460470E-02
	26	0.0000000000000000E 00	0.137755447129237E-03	0.314797829130603E-02
	27	0.0000000000000000E 00	0.166723790595788E-03	0.300153279780958E-02
	28	0.0000000000000000E 00	-0.376280727448811E-04	0.271170305605449E-02
	29	0.0000000000000000E 00	0.196336030232174E-04	0.283756821322627E-02
	30	0.0000000000000000E 00	0.150418669248309E-03	0.278648566520550E-02
	31	0.0000000000000000E 00	0.146044654313119E-03	0.272095483159811E-02
	32	0.0000000000000000E 00	0.646078621666440E-04	0.291753725876869E-02
	33	0.0000000000000000E 00	0.0000000000000000E 00	0.287785366275316E-02
10	1	0.0000000000000000E 00	0.0000000000000000E 00	0.0000000000000000E 00
	2	-0.477075494935557E-04	0.247354695767951E-04	-0.297449774634907E-03
	3	-0.661156731453665E-07	-0.541539628288987E-04	-0.431179402461611E-03
	4	-0.354928990172019E-04	-0.508404354778997E-04	-0.462536520410766E-03
	5	-0.510978931632788E-04	-0.110694291201888E-03	-0.579093877055993E-03
	6	-0.764382232489155E-04	-0.162230422547839E-04	-0.651318688714076E-03
	7	0.651525180326460E-04	-0.509629073834074E-04	-0.786109307845320E-03
	8	0.537734406074020E-04	-0.932575176436813E-04	-0.867159070644456E-03
	9	0.219074163650099E-04	-0.800743564595349E-04	-0.801058845097738E-03
	10	0.140680112635609E-03	-0.164727239022794E-03	-0.910630360588815E-03
	11	0.656501045041260E-04	-0.520655881850533E-04	-0.719750871864923E-03
	12	0.288648558392030E-04	-0.119827402171275E-03	-0.618810759295555E-03
	13	0.872084537880751E-04	-0.122386154302932E-03	-0.310912574085013E-03
	14	0.512351695127705E-04	-0.588967963416902E-04	-0.458811210511811E-04
	15	-0.411010653102058E-05	-0.142967994190588E-03	0.241844610746048E-03
	16	0.239406269108545E-04	-0.493285596378435E-04	0.394311534331206E-03
	17	-0.896276259359519E-04	0.494281352331960E-04	0.577655269591236E-03
	18	0.732016956396687E-04	0.935872155787933E-04	0.887972135312531E-03
	19	0.182662802749422E-04	-0.469844083028907E-04	0.118138338289980E-02
	20	0.393861250870778E-06	0.126356025165336E-04	0.147263315825859E-02
	21	-0.242485902902085E-04	-0.446587975378184E-04	0.170272613889324E-02
	22	-0.112810138387770E-03	0.286144798800673E-04	0.183875867837905E-02
	23	-0.891327464872773E-04	0.484451474457325E-04	0.193448768105694E-02
	24	-0.263679819988676E-04	-0.112796243092430E-03	0.202424906236216E-02
	25	-0.567761624252693E-04	-0.707028815171158E-04	0.193831440419401E-02
	26	-0.960645208063607E-04	-0.289959641679108E-05	0.191755859920860E-02
	27	-0.107932254033565E-03	-0.851920037022304E-04	0.188380301930617E-02
	28	-0.241312513408790E-04	-0.305725262407360E-04	0.189909444888746E-02
	29	-0.102778870261985E-03	0.571390567828722E-04	0.181529237889600E-02
	30	0.119483240745860E-04	-0.894804122827755E-04	0.173078581451851E-02
	31	-0.999146678306900E-04	0.101569886513533E-03	0.196901828242733E-02
	32	-0.439081422556845E-04	0.394949114813401E-04	0.192985816938023E-02
	33	-0.884921111305373E-04	0.0000000000000000E 00	0.178704038348711E-02

Table 5.10 Sample of Cartesian Coordinates and Pressure Arrays at Time Step 400 for M = 2 and 10

TIME STEP 400 TIME 0.16000000E-02

M	N	CARTESIAN COORDINATES Y1(M,N, 400)	Y2(M,N, 400)	Y3(M,N, 400)	PRESSURE P(M,N)
2	1	0.0000000000000000E 00	0.0000000000000000E 00	0.0000000000000000E 00	0.0000000000000000E 00
	2	0.0000000000000000E 00	0.9993716777133409E 00	-0.1262648178627175E 00	0.2485700807166598E-04
	3	0.0000000000000000E 00	0.1991031583460056E 01	-0.2323616051985924E 00	0.2416569194982430E-04
	4	0.0000000000000000E 00	0.2986489925456179E 01	-0.3783307401836761E 00	0.2353134277209950E-04
	5	0.0000000000000000E 00	0.3981943222267893E 01	-0.5036247382130053E 00	0.2295189467931800E-04
	6	0.0000000000000000E 00	0.4977279215126816E 01	-0.6278885780324873E 00	0.2242552216846547E-04
	7	0.0000000000000000E 00	0.5973873659621578E 01	-0.7498195417249140E 00	0.2195062958965738E-04
	8	0.0000000000000000E 00	0.6970204679352300E 01	-0.8687279333135340E 00	0.2132584017065104E-04
	9	0.0000000000000000E 00	0.7967066056050100E 01	-0.9841017077096345E 00	0.2114998305246166E-04
	10	0.0000000000000000E 00	0.8964784084262459E 01	-0.1093678575097817E 01	0.2082208333600219E-04
	11	0.0000000000000000E 00	0.9962482990428790E 01	-0.1196440943171498E 01	0.2054133636010593E-04
	12	0.0000000000000000E 00	0.1096164737018491E 02	-0.1292568567481262E 01	0.2030709414614460E-04
	13	0.0000000000000000E 00	0.1196069987603772E 02	-0.1380266888208165E 01	0.2011083274897154E-04
	14	0.0000000000000000E 00	0.1296155315233641E 02	-0.1459438689658245E 01	0.1997611583088846E-04
	15	0.0000000000000000E 00	0.1396168960406734E 02	-0.1530887197758526E 01	0.1987854574231753E-04
	16	0.0000000000000000E 00	0.1496314750395679E 02	-0.1595319604052816E 01	0.1982569959653854E-04
	17	0.0000000000000000E 00	0.1596420592298380E 02	-0.1653251449847307E 01	0.1981704655641544E-04
	18	0.0000000000000000E 00	0.1696605528139823E 02	-0.1706222462739868E 01	0.1985184236780430E-04
	19	0.0000000000000000E 00	0.1796778599080062E 02	-0.1753663331937867E 01	0.1992899749659824E-04
	20	0.0000000000000000E 00	0.1897011181165700E 02	-0.1796563006683662E 01	0.2004691643679511E-04
	21	0.0000000000000000E 00	0.1997216689402501E 02	-0.1834433154015361E 01	0.2020330836226599E-04
	22	0.0000000000000000E 00	0.2097471467596310E 02	-0.1868393878357596E 01	0.2039497396149522E-04
	23	0.0000000000000000E 00	0.2197707201235867E 02	-0.1898383244352068E 01	0.2061758054003874E-04
	24	0.0000000000000000E 00	0.2297965881916999E 02	-0.1924200070565617E 01	0.2086544765608027E-04
	25	0.0000000000000000E 00	0.2398177576110006E 02	-0.1946892060707336E 01	0.2113137849976894E-04
	26	0.0000000000000000E 00	0.2498428952722194E 02	-0.1966539713263535E 01	0.2140658426237269E-04
	27	0.0000000000000000E 00	0.2598635558126021E 02	-0.1984097669059082E 01	0.2168076067234824E-04
	28	0.0000000000000000E 00	0.2698856611799544E 02	-0.1999254854365450E 01	0.2194237318173783E-04
	29	0.0000000000000000E 00	0.2799079851420297E 02	-0.2011879686532295E 01	0.2217919293717096E-04
	30	0.0000000000000000E 00	0.2899301181564056E 02	-0.2022202472731050E 01	0.2237908615251040E-04
	31	0.0000000000000000E 00	0.2999541595949241E 02	-0.2030107855960577E 01	0.2253100153612287E-04
	32	0.0000000000000000E 00	0.3099743394152207E 02	-0.2035092570681502E 01	0.2262603252068125E-04
	33	0.0000000000000000E 00	0.3200000000000000E 02	-0.2036681488392345E 01	0.2265830084763123E-04
10	1	0.8000000000000000E 01	0.0000000000000000E 00	0.0000000000000000E 00	0.0000000000000000E 00
	2	0.7999563977490974E 01	0.9972683769364465E 00	-0.1019446296688460E 00	0.2561981682961846E-04
	3	0.7999712086712909E 01	0.1994580102525932E 01	-0.2033917672831318E 00	0.2489191555914895E-04
	4	0.8000279382965516E 01	0.2991996070208322E 01	-0.3037297981393019E 00	0.2422101852921266E-04
	5	0.8000933761571326E 01	0.3989626588884287E 01	-0.4028601341717271E 00	0.2360479697225321E-04
	6	0.8001980868231910E 01	0.4987182640609776E 01	-0.5003549457274481E 00	0.2304113853679900E-04
	7	0.8003644651410247E 01	0.5985142456867779E 01	-0.5956646029358369E 00	0.2252813266263065E-04
	8	0.8005206092566362E 01	0.6982969745122238E 01	-0.6876265795014543E 00	0.2206405521109350E-04
	9	0.8007325297163890E 01	0.7981535830587295E 01	-0.7736523272206940E 00	0.2164735178476287E-04
	10	0.8009504886209153E 01	0.8980090870271886E 01	-0.8592292741471640E 00	0.2127661905292575E-04
	11	0.8011690810294600E 01	0.9979242960477335E 01	-0.9374560981248174E 00	0.2095058326374837E-04
	12	0.8013961555925172E 01	0.1097833447980587E 02	-0.1010123607985328E 01	0.2066807497491343E-04
	13	0.8016141112411534E 01	0.1197816145756306E 02	-0.1076726606181334E 01	0.2043799888100709E-04
	14	0.8018490031285267E 01	0.1297821270313844E 02	-0.1137200379552373E 01	0.2022929748438191E-04
	15	0.8020463868701781E 01	0.1397853311481140E 02	-0.1192097204185252E 01	0.2007090724200734E-04
	16	0.8022526733123851E 01	0.1497917457007360E 02	-0.1241783766971052E 01	0.1995170586794729E-04
	17	0.8024348048116800E 01	0.1597995096213136E 02	-0.1286970468835966E 01	0.1987044958475506E-04
	18	0.8025943686447051E 01	0.1698082567746380E 02	-0.1327897633304332E 01	0.1982569959655854E-04
	19	0.8027541754670854E 01	0.1798191073335599E 02	-0.1364717373946746E 01	0.1981573766251531E-04
	20	0.8028617792631183E 01	0.1898296550550013E 02	-0.1397734612435461E 01	0.1983847362391964E-04
	21	0.8030161365341550E 01	0.1998414360568736E 02	-0.1427248917172914E 01	0.1989134418841312E-04
	22	0.8030905396357420E 01	0.2098539559123829E 02	-0.1453394068321960E 01	0.1997121629755957E-04
	23	0.8032030358436995E 01	0.2198684234622984E 02	-0.1476578000781553E 01	0.2007429795275850E-04
	24	0.8032768695292153E 01	0.2298808939254823E 02	-0.1496949710[illegible]8888E 01	0.2019607455429349E-04
	25	0.8033256496175932E 01	0.2398956322261444E 02	-0.1515151815096388E 01	0.2033128694897128E-04
	26	0.8033864385934990E 01	0.2499090073740315E 02	-0.1530978867743081E 01	0.2047397135035152E-04
	27	0.8034198177058198E 01	0.2599221203961742E 02	-0.1544449933396688E 01	0.2061758054003874E-04
	28	0.8034760980703831E 01	0.2699333131926417E 02	-0.1555817904898587E 01	0.2075520059650658E-04
	29	0.8034674052310275E 01	0.2799472303612451E 02	-0.1565422973393227E 01	0.2087986642643304E-04
	30	0.8035437568953836E 01	0.2899589898968563E 02	-0.1573032999027969E 01	0.2098496273379595E-04
	31	0.8035243403329555E 01	0.2999732226299396E 02	-0.1578527657924640E 01	0.2106467704594699E-04
	32	0.8035413447059692E 01	0.3099859116475941E 02	-0.1582186082213971E 01	0.2111445272834811E-04
	33	0.8035287096507280E 01	0.3200000000000000E 02	-0.1583462396072416E 01	0.2113137845976894E-04

Table 5.11 Sample of Surface Normal Components Array at Time Step 400 for M = 2 and 10

TIME STEP 400 TIME 0.1600000E-02

SURFACE NORMAL VECTOR COMPONENTS

M	N	SN1(M,N)	SN2(M,N)	SN3(M,N)
2	1	0.0000000000000000E 00	0.1259438236377535E 00	0.9920373749448670E 00
	2	0.0000000000000000E 00	0.1257431427542407E 00	0.9920628317048709E 00
	3	0.0000000000000000E 00	0.1255925629204617E 00	0.9920819059629451E 00
	4	0.0000000000000000E 00	0.1252118350545883E 00	0.9921300299664581E 00
	5	0.0000000000000000E 00	0.1243460511827116E 00	0.9922389125383393E 00
	6	0.0000000000000000E 00	0.1226627383524021E 00	0.9924484130774204E 00
	7	0.0000000000000000E 00	0.1200987826666495E 00	0.9927726976092886E 00
	8	0.0000000000000000E 00	0.1167375176263626E 00	0.9931628023533890E 00
	9	0.0000000000000000E 00	0.1120704998096157E 00	0.9937002581626024E 00
	10	0.0000000000000000E 00	0.1058160330552835E 00	0.9943857235240474E 00
	11	0.0000000000000000E 00	0.9911081021817243E-01	0.9950764027439789E 00
	12	0.0000000000000000E 00	0.9160816448660351E-01	0.9957951316407395E 00
	13	0.0000000000000000E 00	0.8315004582592052E-01	0.9965370398897480E 00
	14	0.0000000000000000E 00	0.7506055834267283E-01	0.9971789772058405E 00
	15	0.0000000000000000E 00	0.6773044949466867E-01	0.9977035656508910E 00
	16	0.0000000000000000E 00	0.6099148543176072E-01	0.9981382863635817E 00
	17	0.0000000000000000E 00	0.5528623890943591E-01	0.9984705462792825E 00
	18	0.0000000000000000E 00	0.5005440718302291E-01	0.9987464925202773E 00
	19	0.0000000000000000E 00	0.4503310742402104E-01	0.9989854950076784E 00
	20	0.0000000000000000E 00	0.4026297035756706E-01	0.9991891178440573E 00
	21	0.0000000000000000E 00	0.3580995034336639E-01	0.9993586180427953E 00
	22	0.0000000000000000E 00	0.3188060268951205E-01	0.9994916843936990E 00
	23	0.0000000000000000E 00	0.2782353947274756E-01	0.9996128503831914E 00
	24	0.0000000000000000E 00	0.2419041776303528E-01	0.9997073690277820E 00
	25	0.0000000000000000E 00	0.2111620937046566E-01	0.9997770279894290E 00
	26	0.0000000000000000E 00	0.1855710670578592E-01	0.9998278020692916E 00
	27	0.0000000000000000E 00	0.1632049397970176E-01	0.9998668118685930E 00
	28	0.0000000000000000E 00	0.1385888760395115E-01	0.9999039610054462E 00
	29	0.0000000000000000E 00	0.1144761094095897E-01	0.9999344739550409E 00
	30	0.0000000000000000E 00	0.9092715383698147E-02	0.9999586604099945E 00
	31	0.0000000000000000E 00	0.6430697034112497E-02	0.9999793228540536E 00
	32	0.0000000000000000E 00	0.3279282361539989E-02	0.9999946231391413E 00
	33	0.0000000000000000E 00	0.0000000000000000E 00	0.1000000000000000E 01
10	1	0.0000000000000000E 00	0.1019415412234536E 00	0.9947903910739121E 00
	2	-0.5643700761622574E-02	0.1014437297862687E 00	0.9948252702503917E 00
	3	-0.1143812664749513E-01	0.1006425423177331E 00	0.9948568982193464E 00
	4	-0.1736998395783445E-01	0.9948132439608992E-01	0.9948878076213138E 00
	5	-0.2335073289916596E-01	0.9806647370523801E-01	0.9949058799746275E 00
	6	-0.2917157506688620E-01	0.9607055304207587E-01	0.9949469674541989E 00
	7	-0.3497292302421939E-01	0.9342370123232040E-01	0.9950120133461687E 00
	8	-0.4102808415902189E-01	0.8989056789549705E-01	0.9951062163025031E 00
	9	-0.4694960672765971E-01	0.8561549222968173E-01	0.9952214436588410E 00
	10	-0.5258325374939417E-01	0.8073284801207883E-01	0.9953477827759427E 00
	11	-0.5789988963237005E-01	0.7530118006662580E-01	0.9954785034644958E 00
	12	-0.6298870339570264E-01	0.6950597019779783E-01	0.9955909518227811E 00
	13	-0.6763709120161888E-01	0.6342168184504954E-01	0.9956921784194797E 00
	14	-0.7180309514265984E-01	0.5758492549537625E-01	0.9957551365087836E 00
	15	-0.7559503674216125E-01	0.5219804655082300E-01	0.9957713608882089E 00
	16	-0.7908747727443144E-01	0.4735468494097715E-01	0.9957422883218866E 00
	17	-0.8238978550265740E-01	0.4297591378091432E-01	0.9956731361644351E 00
	18	-0.8530158412225167E-01	0.3880057981875988E-01	0.9955993947125331E 00
	19	-0.8797728368294343E-01	0.3484216052599513E-01	0.9955129438664531E 00
	20	-0.9027237920854672E-01	0.3120622659290592E-01	0.9954280832756631E 00
	21	-0.9215066123071945E-01	0.2777219183202221E-01	0.9953577050465760E 00
	22	-0.9397364425234159E-01	0.2460251474355664E-01	0.9952706518601999E 00
	23	-0.9529374439535324E-01	0.2173305518367674E-01	0.9952119254667716E 00
	24	-0.9635656945744200E-01	0.1922648531231980E-01	0.9951602014517759E 00
	25	-0.9744707574357098E-01	0.1696068707256063E-01	0.9950961775240891E 00
	26	-0.9833780519372098E-01	0.1460344726292553E-01	0.9950459363254145E 00
	27	-0.9889205959605420E-01	0.1238696191573141E-01	0.9950210669746390E 00
	28	-0.9949060037864559E-01	0.1044447584608925E-01	0.9949836849220726E 00
	29	-0.9992156550623991E-01	0.8586928754512229E-02	0.9949582631454131E 00
	30	-0.1005023056283959E 00	0.6539137951687736E-02	0.9949153361151389E 00
	31	-0.1009093682423629E 00	0.4545797566577934E-02	0.9949852371632677E 00
	32	-0.1011394037037222E 00	0.2453722013247306E-02	0.9948641534533077E 00
	33	-0.1013195158684104E 00	0.0000000000000000E 00	0.9948539368691819E 00

Table 5.12 Sample of Stress Subincrement Array at Time Step 400 for Layers K = 3 and 4

LMAT(M,N, 3)

N	M= 2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
9	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
12	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
13	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
14	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
17	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
18	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
19	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
20	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
21	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
22	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
23	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
24	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
25	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
26	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
27	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
28	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
29	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
30	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
31	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
32	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
33	[illegible]	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

LMAT(M,N, 4)

N	M= 2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10	0	0	0	1	0	0	1	0	0	0	0	1	1	0	0	0	0	0	0	0	0
11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
12	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
13	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
14	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
15	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
16	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	1	0	0	0	0	0
17	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
18	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
19	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
20	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
21	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
22	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
23	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
24	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
25	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
26	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
27	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
28	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
29	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
30	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
31	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
32	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
33	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Table 5.13 Samples of Energy Balance and of Surface Strains Printed Output at Time Steps 200 and 400

TIME STEP 200 TIME= 0.80000000E-03

SURFACE STRAINS					STRAIN GAGE READING					
ETA1	ETA2	M	N	FACE	ANGLE 0	ANGLE 90	ANGLE		ANGLE	
0.000	32.000	2.000	33.000	OUTER	0.51017709E-03	-0.37827347E-02	45.00	-0.16339714E-02	22.50	-0.11735333E-03
0.000	32.000	2.000	33.000	INNER	0.72733858E-02	0.54911106E-02	45.00	0.63826427E-02	22.50	0.70125748E-02
0.000	0.000	2.000	1.000	OUTER	0.00000000E 00	0.10399005E-02	45.00	0.52008536E-03	22.50	0.15235748E-03
0.000	0.000	2.000	1.000	INNER	0.00000000E 00	0.22662113E-02	45.00	0.11337469E-02	22.50	0.33215983E-03
20.000	32.000	22.000	33.000	OUTER	0.34576470E-02	0.00000000E 00	45.00	0.17303154E-02	22.50	0.29520314E-02
20.000	32.000	22.000	33.000	INNER	0.37381704E-02	0.00000000E 00	45.00	0.18708287E-02	22.50	0.31915986E-02

TIME STEP 200 TIME= 0.80000000E-03 KINETIC= 0.84687997E 06 ELASTIC= 0.34996610E 06 PLASTIC= 0.60309890E 05
TOTAL ENERGY= 0.12571560E 07

TAPE 1 WRITTEN, NCYCLE= 200 TIME= 0.80000000E-03

TIME STEP 400 TIME= 0.16000000E-02

SURFACE STRAINS					STRAIN GAGE READING					
ETA1	ETA2	M	N	FACE	ANGLE 0	ANGLE 90	ANGLE		ANGLE	
0.000	32.000	2.000	33.000	OUTER	-0.14012169E-02	0.97995294E-03	45.00	-0.20992306E-03	22.50	-0.10521479E-02
0.000	32.000	2.000	33.000	INNER	0.14749341E-01	0.41496587E-02	45.00	0.94634127E-02	22.50	0.13203984E-01
0.000	0.000	2.000	1.000	OUTER	0.00000000E 00	0.30641016E-02	45.00	0.15332226E-02	22.50	0.44531381E-03
0.000	0.000	2.000	1.000	INNER	0.00000000E 00	0.33674654E-02	45.00	0.16851478E-02	22.50	0.49386227E-03
20.000	32.000	22.000	33.000	OUTER	0.64226842E-02	0.00000000E 00	45.00	0.32164819E-02	22.50	0.54846680E-02
20.000	32.000	22.000	33.000	INNER	0.51591539E-02	0.00000000E 00	45.00	0.25828955E-02	22.50	0.44052656E-02

TIME STEP 400 TIME= 0.16000000E-02 KINETIC= 0.93843398E 05 ELASTIC= 0.68588570E 06 PLASTIC= 0.47751912E 06
TOTAL ENERGY= 0.12572482E 07

TAPE 1 WRITTEN, NCYCLE= 400 TIME= 0.16000000E-02

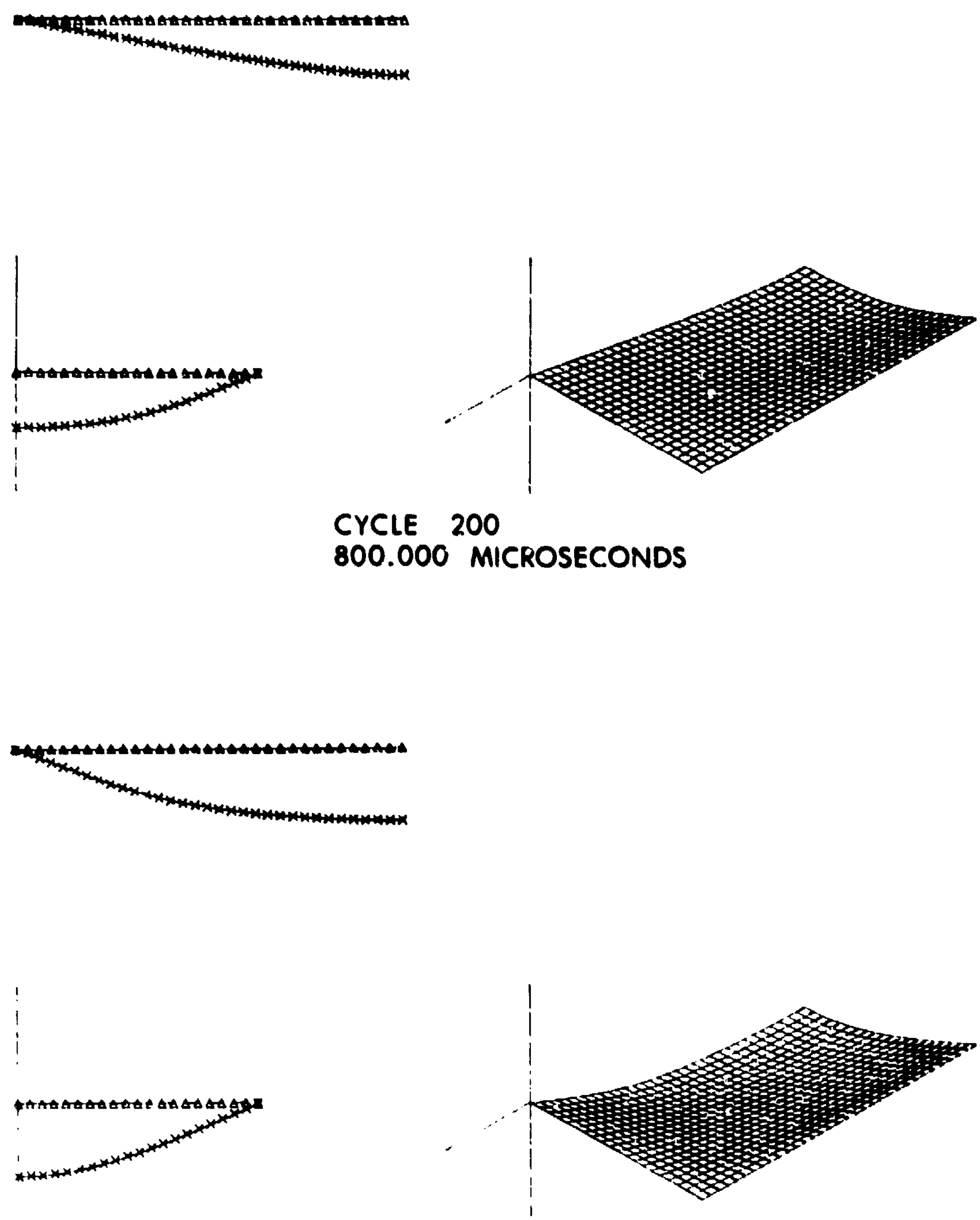


Figure 5.3 Isometric and Cross-sectional Cal Comp Plots of the Deformed Middle Surface at Time Steps 200 and 400 with Displacements Magnified by Factor of Three

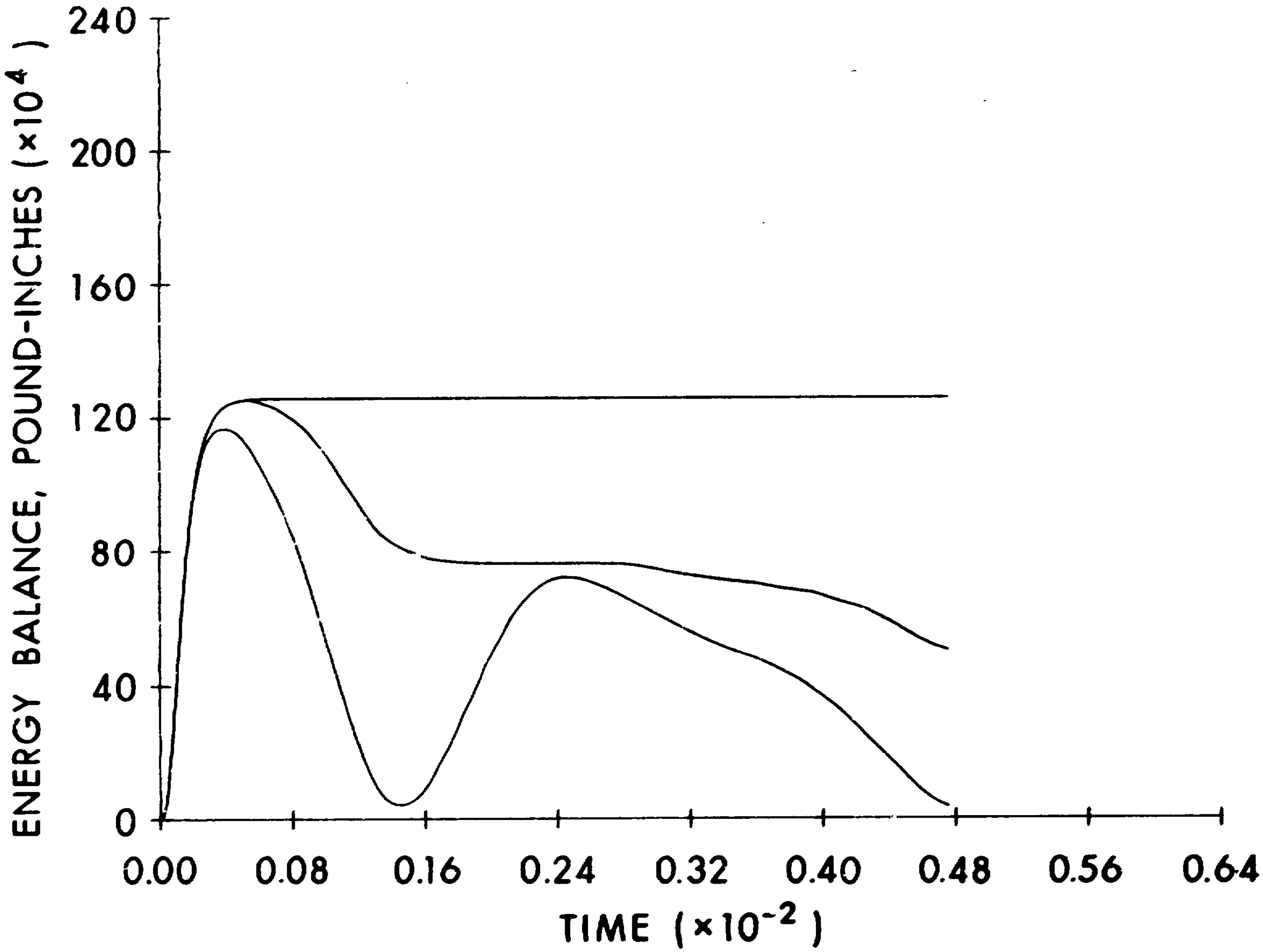


Figure 5.4 Cal Comp Plot of the History of the Energy Balance, Showing in Ascending Order the Kinetic Energy, Total Energy and External Work

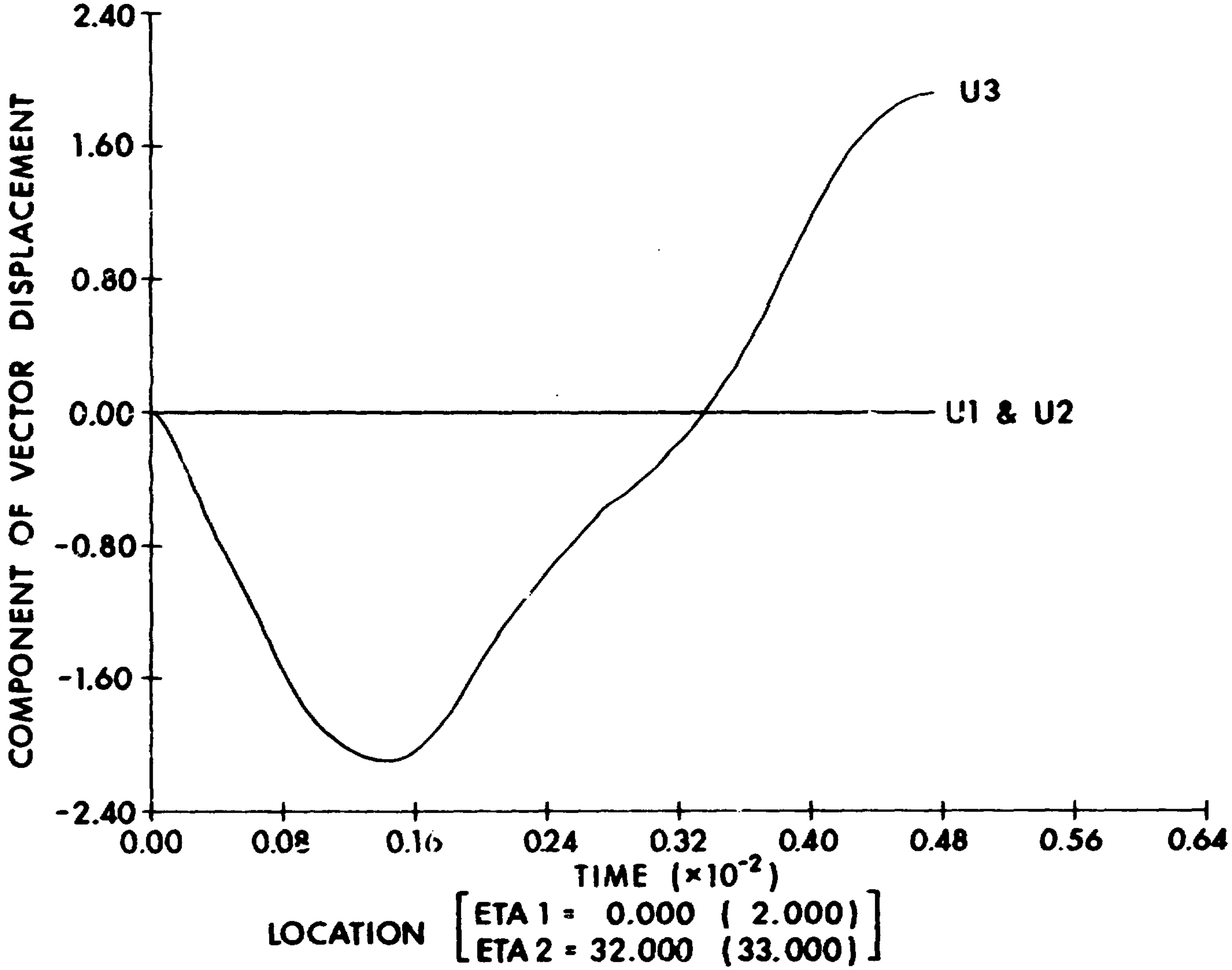
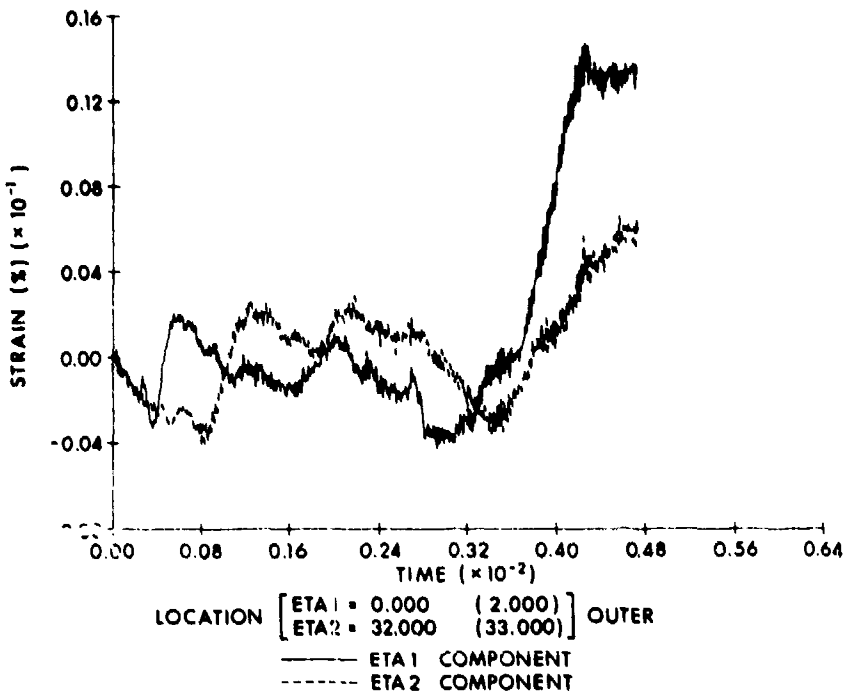


Figure 5.? Cal Comp Plot of the History of the Deflection at the Center of the Plate



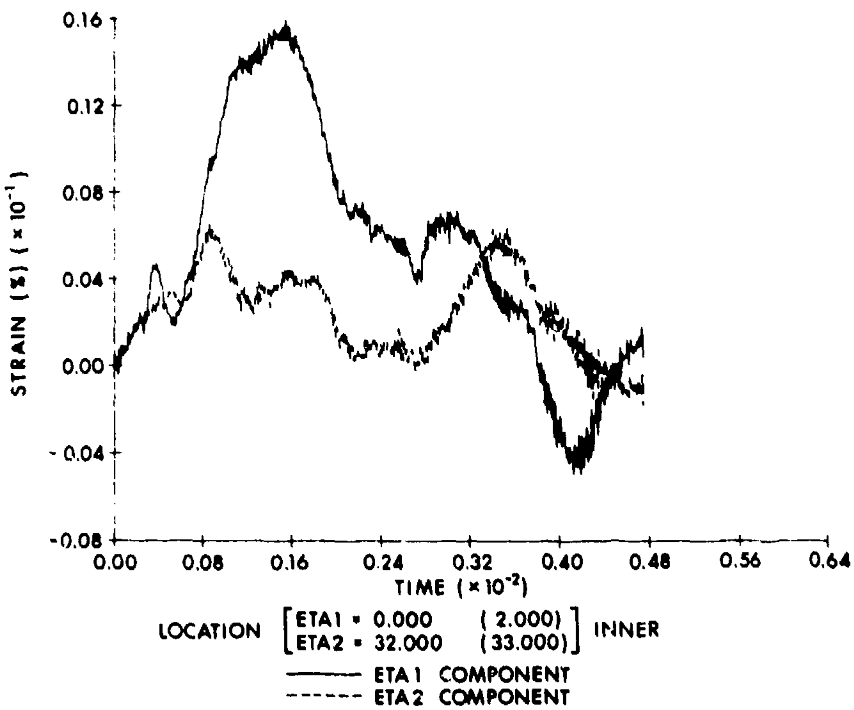


Figure 5.6 Cal Comp Plots of the History of the Surface Elongational Strains Along the Mesh Directions at the Center of Plate

5.2 Example 2: Impulsively Loaded Cylinder

The second example involves a dynamic buckling problem: determine the final equilibrium configuration of a clamped end, aluminum cylinder due to an inwardly directed, instantaneous impulse delivered over a rectangular region of its surface. The impulse imparts a uniform velocity of 7500 in/sec over an 180° arc extending the entire length of the shell. The dimensions and geometry of the problem are shown in Figure 5.7. The example employs the following REPSIL options.

- Cylindrical shell initial geometry
- Input card specified impulsive loading
- Clamped edge and symmetry edge boundaries
- Elastic-perfectly plastic, strain rate insensitive material behavior
- Damping procedure

The material properties of the aluminum are:

Young's Modulus $E = 10.7 \times 10^6$ psi

Poisson's Ratio $\nu = 1/3$

Yield Stress $\sigma_o = 42{,}000$ psi

Mass Density $\rho = 2.59066 \times 10^{-4} \dfrac{\text{lb - sec}^2}{\text{in}^4}$

As already indicated, the material is assumed perfectly plastic at the yield stress and independent of the strain rate.

Advantage is taken of the symmetry of the problem to restrict the analysis to the portion of the shell on the positive side of the y^2, y^3 coordinate plane between the y^1, y^3 coordinate plane and the crossectional symmetry plane, located midway between the ends of the cylinder, Figure 5.7. This makes edges 1, 2 and 3 symmetry boundaries and edge 4 a clamped boundary (compare with Figures 3.1 and 3.7). Also, only 180 degrees of the circumference and half the length are prescribed as the input dimensions and the loaded region is restricted to the first 90 degrees arc. The example uses 20 mesh intervals in the circumferential direction and 12 in the axial direction, giving an almost square mesh. The thickness is divided into 4 layers. The time increment is deliberately set equal to zero in order to assure that the stable time increment determined by the program is used to solve the example.

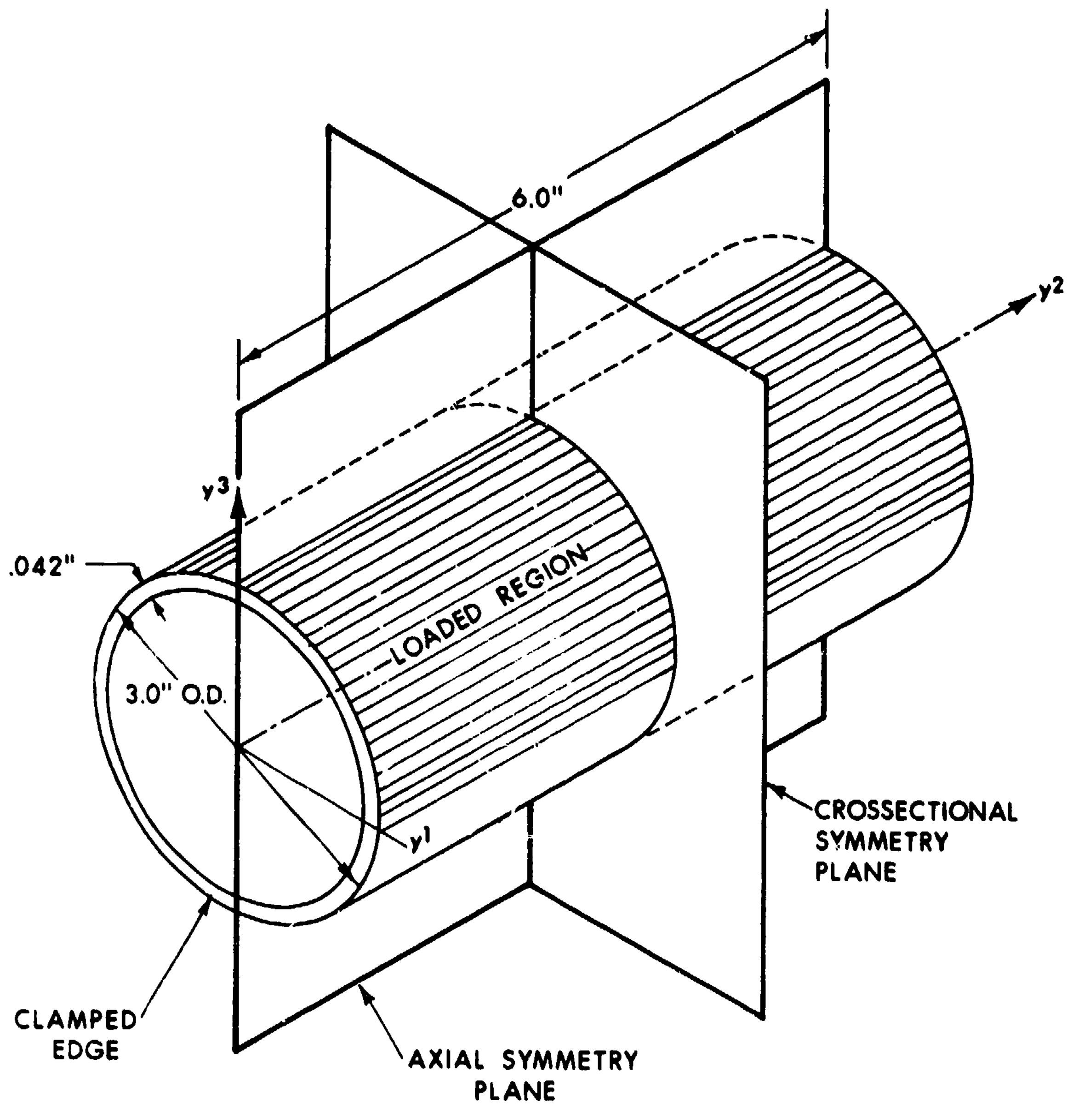


Figure 5.7 Geometry for Example Problem 2

The damping option is used to obtain a final equilibrium configuration. Based on a preliminary solution of the example without the use of damping it is determined that most of the plastic work is accomplished by time step 270. Hence, the damping for this example is picked to begin at time step 270. The example is set to run until time step 600. Since the run may not be terminated by damping before time step 600, provision is made for collecting information for a restart run every 100 time steps. Table 5.14 lists the input data for the run in the order shown in Table 3.1. Notice that while the diameter to the outer surface is specified in Figure 5.9, Card 14 calls for the radius to the middle surface. Also, because half the area associated with each mesh point along the 90° line (i.e. along M = 12) receives the full impulse, only half the impulse velocity is assigned to these points in Cards 16a - 16ℓ.

Tables 5.15 - 5.23 give sample listings of the printed output. For the sake of economy, only the values of arrays Ui(M,N), Yi(M,N), SNi(M,N) at M = 2, 12, 22 and $1 \leq N \leq 13$ (i.e. along the 0°, 90°, 180° meridians) are given. Also, the LMAT (M,N,K) array is given at K = 1,2. Figures 5.8 - 5.11 give examples of the plotted output. Notice that the damping operations terminate the run at time step 525, corresponding to approximately 643 microseconds.

Table 5.14 Input Data Cards for Cylindrical Shell Problem

Card 1	EXAMPLE 2 FULL CYLINDER WITH IMPULSIVE LOADING												
Card 2	20	12	4	0.0									
Card 3	600	0	100	0.0									
Card 4	2	2	2	1									
Card 5	0	0	270	.100000E 00	.500000E-02								
Card 6	.107000E 08	.33333333	.420000E 05	.259066E-03	.420000E-01	1	0						
Card 8	50	1	1	1									
Card 9	7	100	125	200	250	300	400	500					
Card 10	4	100	200	300	400								
Card 11a	24	25	50	75	100	125	150	175	200	225	250	275	300
Card 11b	325	350	375	400	425	450	475	500	525	550	575	600	
Card 12	90.0	1.5	6										
Card 13a	0.0	3.0	45.0	135.0	1								
Card 13b	0.0	3.0	45.0	135.0	0								
Card 13c	45.0	3.0	45.0	135.0	1								
Card 13d	45.0	3.0	45.0	135.0	0								
Card 13e	90.0	1.5	45.0	135.0	1								
Card 13f	90.0	1.5	45.0	135.0	0								
Card 14	.300000E 01	.295800E 01	.360000E 03										
Card 15	2	11	2	13	.750000E 04	13							
Card 16a	12	2	.375000E 04										
Card 16b	12	3	.375000E 04										
Card 16c	12	4	.375000E 04										
Card 16d	12	5	.375000E 04										
Card 16e	12	6	.375000E 04										
Card 16f	12	7	.375000E 04										
Card 16g	12	8	.375000E 04										
Card 16h	12	9	.375000E 04										
Card 16i	12	10	.375000E 04										
Card 16j	12	11	.375000E 04										
Card 16k	12	12	.375000E 04										
Card 16ℓ	12	13	.375000E 04										

Table 5.15 First Page of Printed Output Summarizing the Input Data and Results of Stable Time Increment Check

```
                        SRI REPSIL CODE

      *** EXAMPLE 2 FULL CYLINDER WITH IMPULSIVE LOADING ***

        20 MESHES IN THE ETA1 DIRECTION    (DETA1 =0.232321E 00)
        12 MESHES IN THE ETA2 DIRECTION    (DETA2 =0.250000E 00)

             BENDING   TIME INCREMENT= 0.689333E-05
             MEMBRANE  TIME INCREMENT= 0.157900E-05
             INPUT     TIME INCREMENT= 0.000000E 00

          TIME INCREMENT USED BY REPSIL= 0.150000E-05

                    YOUNG'S MODULUS =0.107000E 08
  POISSON'S RATIO =0.333333E 00         YIELD STRESS    =0.420000E 05
  MASS DENSITY    =0.259066E-03         THICKNESS       =0.420000E-01

                  START AT TIME STEP     0
                  FINAL  TIME  STEP    600
                  SURFACE STRAINS EVERY   50 TIME STEP
                  RESTART  WRITE  EVERY  100 TIME STEP

           LAYER =      4                 NSTRN   =     6
           LOAD  =      0                 LPRESS  =     0

                     BOUNDARY CONDITIONS
               1/2/3 = CLAMPED/SYMMETRY/HINGED
                       EDGE1 =   2
                       EDGE2 =   2
                       EDGE3 =   2
                       EDGE4 =   1

                  PRINT OPTION CONTROL CARD
                    0/1 = NO PRINT/PRINT
                     1 DISPLACEMENT INCREMENTS
                     1 CARTESIAN COORDINATES, PRESSURE
                     1 SURFACE NORMAL VECTOR COMPONENTS

PRINT INFORMATION AT THE FOLLOWING TIME STEPS
  100  125  200  250  300  400  500
PRINT L MATRIX (LMAT) AT THE FOLLOWING TIME STEPS
  100  200  300  400
3-D PLOTS FOR THE FOLLOWING TIME STEPS
   25   50   75  100  125  150  175  200  225  250  275  300  325  350  375  400
  425  450  475  500  525  550  575  600

CONSTITUTIVE RELATION     ELASTOPLASTIC-NO WORK HARDENING-STRAIN RATE INDEPENDENT

                   STRESS-STRAIN AND STRAIN RATE PARAMETERS
   J          SSIG(J)          SEPS(J)            DSR(J)          1/PSR(J)
   1       4.2000000E 04    3.9252336E-03      0.0000000E 00   0.0000000E 00


                   START DAMPING AFTER TIME STEP  270      TIME =0.4050E-03
          DAMPF =0.1000E 00                    DFACT =0.5000E-02
```

Table 5.10 Second Page of Printed Output Summarizing the Input Impulse Velocities and Sample of Third Page of Output with Initial Values of Cartesian Coordinates Array for M = 2, 12, and 22

(M= 2, 11) AND (N= 2, 13) RECEIVE FULL VELOCITY,(VH)= 0.750000E 04

OTHER VELOCITY DISTRIBUTION

M	N	V
12	2	0.375000E 04
12	3	0.375000E 04
12	4	0.375000E 04
12	5	0.375000E 04
12	6	0.375000E 04
12	7	0.375000E 04
12	8	0.375000E 04
12	9	0.375000E 04
12	10	0.375000E 04
12	11	0.375000E 04
12	12	0.375000E 04
12	13	0.375000E 04

INITIAL CARTESIAN COORDINATES

M	N	Y1(M,N)	Y2(M,N)	Y3(M,N)
2	1	0.0000000000000000E 00	0.0000000000000000E 00	0.1479000000000000E 01
	2	0.0000000000000000E 00	0.2500000000000000E 00	0.1479000000000000E 01
	3	0.0000000000000000E 00	0.5000000000000000E 00	0.1479000000000000E 01
	4	0.0000000000000000E 00	0.7500000000000000E 00	0.1479000000000000E 01
	5	0.0000000000000000E 00	0.1000000000000000E 01	0.1479000000000000E 01
	6	0.0000000000000000E 00	0.1250000000000000E 01	0.1479000000000000E 01
	7	0.0000000000000000E 00	0.1500000000000000E 01	0.1479000000000000E 01
	8	0.0000000000000000E 00	0.1750000000000000E 01	0.1479000000000000E 01
	9	0.0000000000000000E 00	0.2000000000000000E 01	0.1479000000000000E 01
	10	0.0000000000000000E 00	0.2250000000000000E 01	0.1479000000000000E 01
	11	0.0000000000000000E 00	0.2500000000000000E 01	0.1479000000000000E 01
	12	0.0000000000000000E 00	0.2750000000000000E 01	0.1479000000000000E 01
	13	0.0000000000000000E 00	0.3000000000000000E 01	0.1479000000000000E 01
12	1	0.1479000000000000E 01	0.0000000000000000E 00	0.1289639377135692E-15
	2	0.1479000000000000E 01	0.2500000000000000E 00	0.1289639377135692E-15
	3	0.1479000000000000E 01	0.5000000000000000E 00	0.1289639377135692E-15
	4	0.1479000000000000E 01	0.7500000000000000E 00	0.1289639377135692E-15
	5	0.1479000000000000E 01	0.1000000000000000E 01	0.1289639377135692E-15
	6	0.1479000000000000E 01	0.1250000000000000E 01	0.1289639377135692E-15
	7	0.1479000000000000E 01	0.1500000000000000E 01	0.1289639377135692E-15
	8	0.1479000000000000E 01	0.1750000000000000E 01	0.1289639377135692E-15
	9	0.1479000000000000E 01	0.2000000000000000E 01	0.1289639377135692E-15
	10	0.1479000000000000E 01	0.2250000000000000E 01	0.1289639377135692E-15
	11	0.1479000000000000E 01	0.2500000000000000E 01	0.1289639377135692E-15
	12	0.1479000000000000E 01	0.2750000000000000E 01	0.1289639377135692E-15
	13	0.1479000000000000E 01	0.3000000000000000E 01	0.1289639377135692E-15
22	1	0.5158557508554767E-15	0.0000000000000000E 00	-0.1479000000000000E 01
	2	0.5158557508554767E-15	0.2500000000000000E 00	-0.1479000000000000E 01
	3	0.5158557508554767E-15	0.5000000000000000E 00	-0.1479000000000000E 01
	4	0.5158557508554767E-15	0.7500000000000000E 00	-0.1479000000000000E 01
	5	0.5158557508554767E-15	0.1000000000000000E 01	-0.1479000000000000E 01
	6	0.5158557508554767E-15	0.1250000000000000E 01	-0.1479000000000000E 01
	7	0.5158557508554767E-15	0.1500000000000000E 01	-0.1479000000000000E 01
	8	0.5158557508554767E-15	0.1750000000000000E 01	-0.1479000000000000E 01
	9	0.5158557508554767E-15	0.2000000000000000E 01	-0.1479000000000000E 01
	10	0.5158557508554767E-15	0.2250000000000000E 01	-0.1479000000000000E 01
	11	0.5158557508554767E-15	0.2500000000000000E 01	-0.1479000000000000E 01
	12	0.5158557508554767E-15	0.2750000000000000E 01	-0.1479000000000000E 01
	13	0.5158557508554767E-15	0.3000000000000000E 01	-0.1479000000000000E 01

Table 5.17 Samples for M = 2, 12, and 22 of Displacement Increments Array During Time Increment 124 to 125 and of Cartesian Coordinates and Pressure Arrays at Time Step 125

DISPLACEMENT INCREMENTS BETWEEN T.S. 124 AND 125

M	N	U1(M,N)	U2(M,N)	U3(M,N)
2	1	0.0000000000000000E 00	0.0000000000000000E 00	0.0000000000000000E 00
	2	0.0000000000000000E 00	-0.4919250473058012E-04	-0.5901174090041111E-04
	3	0.0000000000000000E 00	-0.8038066736741004E-04	-0.7384469436016449E-04
	4	0.0000000000000000E 00	-0.6392270289433926E-04	-0.1229249511509717E-03
	5	0.0000000000000000E 00	-0.8000925049367083E-04	0.2940714170868206E-04
	6	0.0000000000000000E 00	-0.1980599215491332E-04	0.4850418300964931E-03
	7	0.0000000000000000E 00	0.2163460007768018E-04	0.6197591969926490E-03
	8	0.0000000000000000E 00	-0.1339174765439396E-04	0.1001190219630639E-02
	9	0.0000000000000000E 00	-0.2693674494941236E-04	0.1174968214009079E-02
	10	0.0000000000000000E 00	-0.3501429880631098E-04	0.1485919945704606E-02
	11	0.0000000000000000E 00	-0.5987874930071089E-04	0.1930227303659874E-02
	12	0.0000000000000000E 00	-0.4108824274602329E-04	0.2024591023610024E-02
	13	0.0000000000000000E 00	0.0000000000000000E 00	0.2108293177730467E-02
12	1	0.0000000000000000E 00	0.0000000000000000E 00	0.0000000000000000E 00
	2	-0.6713834042915714E-04	-0.6074039779896987E-04	0.1560890993593391E-04
	3	-0.2684253617028086E-03	-0.6410431959494991E-04	-0.1596799211709799E-04
	4	-0.4419920579958609E-03	-0.7654741998882704E-04	-0.7066969676008289E-04
	5	-0.6269399459802745E-03	-0.1659615403277468E-03	-0.8269305499901808E-04
	6	-0.8433503721214100E-03	-0.1931311272220067E-03	-0.9091812439016177E-04
	7	-0.1130811105779498E-02	-0.1898884244470868E-03	-0.2660162103376930E-03
	8	-0.1358304482963970E-02	-0.1824397784034776E-03	-0.2480385830394590E-03
	9	-0.1592171683971822E-02	-0.2924109269873709E-03	-0.1774923211726201E-03
	10	-0.1651791747430002E-02	-0.2079241480215711E-03	-0.2974399544530798E-03
	11	-0.1820124259874403E-02	-0.2019109519349178E-03	-0.3908194330864984E-03
	12	-0.1800936647091604E-02	-0.1281300842414359E-03	-0.4582611351870567E-03
	13	-0.1946178029710765E-02	0.0000000000000000E 00	-0.5323398205204447E-03
22	1	0.0000000000000000E 00	0.0000000000000000E 00	0.0000000000000000E 00
	2	0.0000000000000000E 00	0.3417682793621011E-09	-0.1109988179697020E-04
	3	0.0000000000000000E 00	-0.4697711480740312E-04	-0.4780135919648080E-04
	4	0.0000000000000000E 00	-0.1057995713679926E-03	0.1146439367217694E-03
	5	0.0000000000000000E 00	-0.2265414583922089E-04	0.5413483584198748E-03
	6	0.0000000000000000E 00	-0.3077458897810546E-04	0.1148460647514958E-02
	7	0.0000000000000000E 00	-0.1224113571563953E-03	0.1481961910180744E-02
	8	0.0000000000000000E 00	-0.1027160211048859E-03	0.1595469712074468E-02
	9	0.0000000000000000E 00	-0.2995250283450309E-04	0.1758407913761079E-02
	10	0.0000000000000000E 00	-0.8416094302018707E-04	0.1831093778767649E-02
	11	0.0000000000000000E 00	-0.7985100991164618E-04	0.1934990765453120E-02
	12	0.0000000000000000E 00	-0.1115278127045637E-04	0.2148958569216324E-02
	13	0.0000000000000000E 00	0.0000000000000000E 00	0.2319140290833394E-02

TIME STEP 125 TIME 0.18750000E-03

		CARTESIAN COORDINATES			PRESSURE
M	N	Y1(M,N, 125)	Y2(M,N, 125)	Y3(M,N, 125)	P(M,N)
2	1	0.0000000000000000E 00	0.0000000000000000E 00	0.1479000000000000E 01	0.0000000000000000E 00
	2	0.0000000000000000E 00	0.2427202714710254E 00	0.1421736773857553E 01	0.0000000000000000E 00
	3	0.0000000000000000E 00	0.4616879754696976E 00	0.1249947095430219E 01	0.0000000000000000E 00
	4	0.0000000000000000E 00	0.7248977815407707E 00	0.1208279190023050E 01	0.0000000000000000E 00
	5	0.0000000000000000E 00	0.9768292391844542E 00	0.1194931170655631E 01	0.0000000000000000E 00
	6	0.0000000000000000E 00	0.1238626941777664E 01	0.1181443882140505E 01	0.0000000000000000E 00
	7	0.0000000000000000E 00	0.1498924027871210E 01	0.1168679974263159E 01	0.0000000000000000E 00
	8	0.0000000000000000E 00	0.1743282033043640E 01	0.1159013445143616E 01	0.0000000000000000E 00
	9	0.0000000000000000E 00	0.1997163806697489E 01	0.1157077244789973E 01	0.0000000000000000E 00
	10	0.0000000000000000E 00	0.2246620980885431E 01	0.1150250984406146E 01	0.0000000000000000E 00
	11	0.0000000000000000E 00	0.2501200494784020E 01	0.1152912043729297E 01	0.0000000000000000E 00
	12	0.0000000000000000E 00	0.2746271767926553E 01	0.1151308087006174E 01	0.0000000000000000E 00
	13	0.0000000000000000E 00	0.3000000000000000E 01	0.1151887486544421E 01	0.0000000000000000E 00
12	1	0.1479000000000000E 01	0.0000000000000000E 00	0.1209630177149692E-15	0.0000000000000000E 00
	2	0.1471237108239441E 01	0.2550141249148340E 00	-0.1310262758439254E-01	0.0000000000000000E 00
	3	0.1447948432957640E 01	0.5115940288035692E 00	-0.3869109409548211E-01	0.0000000000000000E 00
	4	0.1380130418269254E 01	0.7497311289422440E 00	-0.9248911192740159E-01	0.0000000000000000E 00
	5	0.1309359087126191E 01	0.9901210011473038E 00	-0.1445252463399384E 00	0.0000000000000000E 00
	6	0.1257983933495613E 01	0.1231967330909959E 01	-0.1906217859429035E 00	0.0000000000000000E 00
	7	0.1209104501967043E 01	0.1478605170805791E 01	-0.2277246408433100E 00	0.0000000000000000E 00
	8	0.1175432985101092E 01	0.1729758274181429E 01	-0.2584149833043266E 00	0.0000000000000000E 00
	9	0.1142433530926913E 01	0.1975052809889107E 01	-0.2854948131879307E 00	0.0000000000000000E 00
	10	0.1118944941052716E 01	0.2226960717815735E 01	-0.3079007431005364E 00	0.0000000000000000E 00
	11	0.109831[illegible]93635068E 01	0.2480202739306824E 01	-0.3272085834977743E 00	0.0000000000000000E 00
	12	0.1085515491314728E 01	0.2739110028630017E 01	-0.3390269981196616E 00	0.0000000000000000E 00
	13	0.1080659069311803E 01	0.3000000000000000E 01	-0.3426805708911055E 00	0.0000000000000000E 00
22	1	0.5158557508554767E-15	0.0000000000000000E 00	-0.1479000000000000E 01	0.0000000000000000E 00
	2	0.5158557508554767E-15	0.2518608950203314E 00	-0.1473454782011211E 01	0.0000000000000000E 00
	3	0.5158557508554767E-15	0.5023338077320956E 00	-0.1456815128044945E 01	0.0000000000000000E 00
	4	0.5158557508554767E-15	0.7522100998569182E 00	-0.1439145832914213E 01	0.0000000000000000E 00
	5	0.5158557508554767E-15	0.1002277807614370E 01	-0.1418953799186472E 01	0.0000000000000000E 00
	6	0.5158557508554767E-15	0.1252097659532654E 01	-0.1402288247999200E 01	0.0000000000000000E 00
	7	0.5158557508554767E-15	0.1501991690004186E 01	-0.1388702913674729E 01	0.0000000000000000E 00
	8	0.5158557508554767E-15	0.1791723966756799E 01	-0.1376953268274813E 01	0.0000000000000000E 00
	9	0.5158557508554767E-15	0.2001492230894970E 01	-0.1366014089206527E 01	0.0000000000000000E 00
	10	0.5158557508554767E-15	0.2250813123436625E 01	-0.1355459519257903E 01	0.0000000000000000E 00
	11	0.5158557508554767E-15	0.2500489889250661E 01	-0.1342429557707983E 01	0.0000000000000000E 00
	12	0.5158557508554767E-15	0.2790149476058559E 01	-0.1329016708852108E 01	0.0000000000000000E 00
	13	0.5158557508554767E-15	0.3000000000000000E 01	-0.1324394094273664E 01	0.0000000000000000E 00

Table 5.18 Samples at Time Step 125 of Surface Normal Components Array for M = 2, 12, and 22 and of Stress Subincrement Array for Layers K = 1 and 2

TIME STEP 125   TIME 0.18750000E-03   SURFACE NORMAL VECTOR COMPONENTS

M	N	SN1(M,N)	SN2(M,N)	SN3(M,N)
2	1	0.0000000000000000E 00	0.0000000000000000E 00	0.1000000000000000E 01
	2	0.0000000000000000E 00	0.4444312447793719E 00	0.8958129457242470E 00
	3	0.0000000000000000E 00	0.4070728271029699E 00	0.9129942043243015E 00
	4	0.0000000000000000E 00	0.1041934606428407E 00	0.9943494606493115E 00
	5	0.0000000000000000E 00	0.4824024704507269E-01	0.9988357415983900E 00
	6	0.0000000000000000E 00	0.5121084050940085E-01	0.9986874537945847E 00
	7	0.0000000000000000E 00	0.4444220470193581E-01	0.9990119544344128E 00
	8	0.0000000000000000E 00	0.3242610953039747E-01	0.9994676246192216E 00
	9	0.0000000000000000E 00	0.1740649901423086E-01	0.9998484954192146E 00
	10	0.0000000000000000E 00	-0.1696224960430659E-02	0.9999986284459649E 00
	11	0.0000000000000000E 00	-0.2114902456378303E-02	0.9999977414720934E 00
	12	0.0000000000000000E 00	0.2594197467220503E-02	0.9999970902103243E 00
	13	0.0000000000000000E 00	0.0000000000000000E 00	0.1000000000000000E 01
12	1	0.1000000000000000E 01	0.0000000000000000E 00	0.0000000000000000E 00
	2	0.9988119822437949E 00	0.7014094827682427E-01	0.1333639809233363E 00
	3	0.8194441259686930E 00	0.2390420189961810E 00	0.5206087409649699E 00
	4	0.5926773201473369E 00	0.3334921389067471E 00	0.7331952279865795E 00
	5	0.4122609363192127E 00	0.2807044209802338E 00	0.8667193013015949E 00
	6	0.3372981302632968E 00	0.2249100953430971E 00	0.9141696351631175E 00
	7	0.2758761905768527E 00	0.1799964716980590E 00	0.9449692719472111E 00
	8	0.2453725781617717E 00	0.1449019333205072E 00	0.9585987101967861E 00
	9	0.2304114815864573E 00	0.1193394034313721E 00	0.9707292915657131E 00
	10	0.1939305140729541E 00	0.9791024480719161E-01	0.9769345945175506E 00
	11	0.1917331534485995E 00	0.7146707751008013E-01	0.9807473681910493E 00
	12	0.1517328960837974E 00	0.3542301074904544E-01	0.9827097842913494E 00
	13	0.1638591051133647E 00	0.0000000000000000E 00	0.9824533882961762E 00
22	1	0.0000000000000000E 00	0.0000000000000000E 00	-0.1000000000000000E 01
	2	0.0000000000000000E 00	0.4411260021811249E-01	-0.9990245628142639E 00
	3	0.0000000000000000E 00	0.4873496796839067E-01	-0.9976605878414849E 00
	4	0.0000000000000000E 00	0.7592302594402370E-01	-0.9971439479121150E 00
	5	0.0000000000000000E 00	0.7397103234392299E-01	-0.9972899247772691E 00
	6	0.0000000000000000E 00	0.6042479737221211E-01	-0.9981727959145771E 00
	7	0.0000000000000000E 00	0.5064979099790474E-01	-0.9987168365981347E 00
	8	0.0000000000000000E 00	0.4377940797971921E-01	-0.9990412290901704E 00
	9	0.0000000000000000E 00	0.4372400947241024E-01	-0.9990739408864730E 00
	10	0.0000000000000000E 00	0.4881108652730347E-01	-0.9988079864227111E 00
	11	0.0000000000000000E 00	0.5298223421255577E-01	-0.9986007987101525E 00
	12	0.0000000000000000E 00	0.3670063723863258E-01	-0.9993480818307120E 00
	13	0.0000000000000000E 00	0.0000000000000000E 00	-0.1000000000000000E 01

TIME STEP 125   TIME 1.8750000E-04,   SUBDIVISIONS OF TIME INCREMENT IN STRESS

LMAT(M,N, 1)

N \ M=	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
3	0	0	0	0	0	1	0	1	1	1	0	0	0	0	0	0	0	0	0	0	0
4	0	0	0	1	1	1	0	0	0	1	0	1	0	0	0	0	0	0	0	0	1
5	0	0	0	1	1	1	0	0	1	0	1	1	0	0	0	0	0	0	0	0	1
6	1	1	1	0	1	1	0	1	1	0	0	0	0	0	0	0	0	0	0	0	1
7	0	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	1
8	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	1	1
9	1	1	1	1	1	0	1	1	1	1	1	0	0	0	1	1	0	0	0	0	1
10	1	1	0	1	0	1	1	1	1	1	1	1	1	0	1	1	0	0	0	1	1
11	1	1	0	0	0	1	1	1	1	1	1	1	0	0	0	1	0	0	0	1	1
12	1	1	0	0	0	1	1	1	1	1	1	1	0	0	0	1	0	0	0	1	1
13	1	1	0	0	0	1	1	1	1	1	1	0	0	0	0	1	0	0	0	0	1

LMAT(M,N, 2)

N \ M=	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
3	0	0	0	0	0	1	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0
4	0	0	0	1	1	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
5	0	0	0	0	0	1	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0
6	0	0	1	0	0	1	0	1	1	0	1	0	0	0	0	0	0	0	0	0	0
7	0	0	0	1	1	1	1	1	1	0	1	0	0	0	0	0	0	0	0	0	0
8	0	0	1	0	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0
9	0	0	1	1	1	0	1	1	1	1	0	0	0	0	1	0	0	0	0	0	0
10	1	0	0	0	0	1	1	1	1	1	1	1	0	0	1	0	0	0	0	0	0
11	0	0	0	0	0	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0
12	1	0	0	0	0	1	1	1	1	1	1	1	0	0	0	1	0	0	0	0	1
13	1	0	0	0	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	1

Table 5.19 Samples for M = 2, 12, and 22 of Displacement Increments Array During Time Increment 249 to 250 and of Cartesian Coordinates and Pressure Arrays at Time Step 250

DISPLACEMENT INCREMENTS BETWEEN T.S. 249 AND 250

M	N	U1(M,N)	U2(M,N)	U3(M,N)
2	1	0.0000000000000000E 00	0.0000000000000000E 00	0.0000000000000000E 00
	2	0.0000000000000000E 00	-0.4969592154502772E-05	0.7590279110993970E-04
	3	0.0000000000000000E 00	0.1014170941296706E-03	0.3030341647998144E-03
	4	0.0000000000000000E 00	0.1173073096767086E-03	0.2592273903244670E-03
	5	0.0000000000000000E 00	0.1061177318086420E-03	0.3440954323919007E-04
	6	0.0000000000000000E 00	0.1707000400029961E-03	-0.2091206576543098E-04
	7	0.0000000000000000E 00	0.2309279020785069E-03	-0.4049470634602093E-04
	8	0.0000000000000000E 00	0.1096795603440901E-03	-0.8067379321923792E-03
	9	0.0000000000000000E 00	0.1071909902490590E-03	-0.1094409643159739E-02
	10	0.0000000000000000E 00	0.1670244013007159E-03	-0.1318706954470879E-02
	11	0.0000000000000000E 00	0.7297690059757090E-04	-0.1403069042404940E-02
	12	0.0000000000000000E 00	0.2909477970243445E-04	-0.1509173039933204E-02
	13	0.0000000000000000E 00	0.0000000000000000E 00	-0.1587901102290927E-02
12	1	0.0000000000000000E 00	0.0000000000000000E 00	0.0000000000000000E 00
	2	0.7709751510622002E-04	0.9012930361029031E-05	-0.6724761523972060E-04
	3	0.3103900647449.49E-03	0.1314279213996079E-04	0.3402030907239109E-04
	4	0.4712635199240577E-03	0.9945273080023649E-04	0.4570903137132709E-04
	5	0.4597708580740920E-03	0.9029117730231300E-04	0.3904364993594079E-04
	6	0.3127200870204091E-03	0.7729107710231194E-04	-0.7212748400761019E-04
	7	0.2100909953024902E-03	-0.9410877101901020E-05	-0.2610956317973230E-03
	8	0.1403906091231271E-03	-0.4921097041315302E-04	-0.4707771602909470E-03
	9	0.7051743906730540E-04	-0.4061344204790991E-04	-0.9097060200012443E-03
	10	0.5415992731913306E-04	0.1029410620976950E-04	-0.9245704476972177E-03
	11	0.3124951375491440E-04	-0.3117604712079447E-04	-0.7130959411001109E-03
	12	0.6400933845771210E-04	-0.4437095640610100E-04	-0.9093984753174197E-03
	13	0.2474193746470542E-04	0.0000000000000000E 00	-0.4703613104190044E-03
22	1	0.0000000000000000E 00	0.0000000000000000E 00	0.0000000000000000E 00
	2	0.0000000000000000E 00	0.6329420970429907E-04	-0.1057441136729351E-03
	3	0.0000000000000000E 00	-0.1960271414747239E-04	-0.4229704940917400E-03
	4	0.0000000000000000E 00	0.1850095072030990E-04	-0.5599040400373090E-03
	5	0.0000000000000000E 00	0.4524005607620928E-04	-0.9204700995470039E-03
	6	0.0000000000000000E 00	-0.6067500033677730E-04	-0.3510729433029570E-03
	7	0.0000000000000000E 00	-0.6625000306210870E-04	-0.9167149901012330E-03
	8	0.0000000000000000E 00	0.7801997991700640E-04	-0.3600747202664330E-03
	9	0.0000000000000000E 00	0.1309732672604503E-03	-0.3374105704107147E-03
	10	0.0000000000000000E 00	-0.9204035393170080E-04	-0.6900063320777271E-03
	11	0.0000000000000000E 00	-0.1149340200920318E-05	-0.1420701300494176E-02
	12	0.0000000000000000E 00	0.1119004634479430E-03	-0.2400950360751990E-02
	13	0.0000000000000000E 00	0.0000000000000000E 00	-0.3034143232090060E-02

TIME STEP 250 TIME 0.1750000E-03

CARTESIAN COORDINATES

M	N	Y1(M,N, 250)	Y2(M,N, 250)	Y3(M,N, 250)	PRESSURE P(M,N)
2	1	0.0000000000000000E 00	0.0000000000000000E 00	0.1479000000000000E 01	0.0000000000000000E 00
	2	0.0000000000000000E 00	0.2415976060670652E 00	0.1470421503912741E 01	0.0000000000000000E 00
	3	0.0000000000000000E 00	0.4577834186118697E 00	0.1244886056509716E 01	0.0000000000000000E 00
	4	0.0000000000000000E 00	0.7218496370059794E 00	0.1203541904600500E 01	0.0000000000000000E 00
	5	0.0000000000000000E 00	0.9726996316959771E 00	0.1203906260130316E 01	0.0000000000000000E 00
	6	0.0000000000000000E 00	0.1235840520476572E 01	0.1203037302732701E 01	0.0000000000000000E 00
	7	0.0000000000000000E 00	0.1484660091907119E 01	0.1206246760049217E 01	0.0000000000000000E 00
	8	0.0000000000000000E 00	0.1739667361127515E 01	0.1212491240927646E 01	0.0000000000000000E 00
	9	0.0000000000000000E 00	0.1993228339671967E 01	0.1217913797043290E 01	0.0000000000000000E 00
	10	0.0000000000000000E 00	0.2243519594077640E 01	0.1225916092600224E 01	0.0000000000000000E 00
	11	0.0000000000000000E 00	0.2498505077096173E 01	0.1232206540209746E 01	0.0000000000000000E 00
	12	0.0000000000000000E 00	0.2745159714180164E 01	0.1234048286966374E 01	0.0000000000000000E 00
	13	0.0000000000000000E 00	0.3000000000000000E 01	0.1275400164661535E 01	0.0000000000000000E 00
12	1	0.1479000000000000E 01	0.0000000000000000E 00	0.1280639377130692E-10	0.0000000000000000E 00
	2	0.1466170159278424E 01	0.2544920115970233E 00	-0.1242129910393301E-01	0.0000000000000000E 00
	3	0.1428512621113607E 01	0.5086349510470085E 00	-0.3730311179912458E-01	0.0000000000000000E 00
	4	0.1359427607073632E 01	0.7493327590011636E 00	-0.8960399694714399E-01	0.0000000000000000E 00
	5	0.1290561530966420E 01	0.9867674644490078E 00	-0.1397702990396041E 00	0.0000000000000000E 00
	6	0.1239394763786087E 01	0.1228690802432108E 01	-0.1823054901004466E 00	0.0000000000000000E 00
	7	0.1192456040339900E 01	0.1479940503191248E 01	-0.2157363359574921E 00	0.0000000000000000E 00
	8	0.1159128403910269E 01	0.1723110308306831E 01	-0.2419313590301290E 00	0.0000000000000000E 00
	9	0.1120210230104851E 01	0.1973048409402318E 01	-0.2633597479440173E 00	0.0000000000000000E 00
	10	0.1105422946811808E 01	0.2225155531767536E 01	-0.2704601300234080E 00	0.0000000000000000E 00
	11	0.1086102042583024E 01	0.2479292790891454E 01	-0.2924075020901359E 00	0.0000000000000000E 00
	12	0.1074160100008935E 01	0.2730671935300010E 01	-0.3013128230669450E 00	0.0000000000000000E 00
	13	0.1070001244455173E 01	0.3000000000000000E 01	-0.3040670020642424E 00	0.0000000000000000E 00
22	1	0.5150557500559767E-15	0.0000000000000000E 00	-0.1479000000000000E 01	0.0000000000000000E 00
	2	0.5150557500559767E-15	0.2500050670703279E 00	-0.1476939609675980E 01	0.0000000000000000E 00
	3	0.5150557500559767E-15	0.5000464130919637E 00	-0.1470054243070235E 01	0.0000000000000000E 00
	4	0.5150557500559767E-15	0.7500697040072913E 00	-0.1463037306437540E 01	0.0000000000000000E 00
	5	0.5150557500559767E-15	0.1000697660438257E 01	-0.1451639311197430E 01	0.0000000000000000E 00
	6	0.5150557500559767E-15	0.1250432379840975E 01	-0.1430922277640975E 01	0.0000000000000000E 00
	7	0.5150557500559767E-15	0.1500239097093635E 01	-0.1424773049605449E 01	0.0000000000000000E 00
	8	0.5150557500559767E-15	0.1750052090020509E 01	-0.1411744761356635E 01	0.0000000000000000E 00
	9	0.5150557500559767E-15	0.2000112741600268E 01	-0.1400017600296737E 01	0.0000000000000000E 00
	10	0.5150557500559767E-15	0.2249970091628624E 01	-0.1396905980407825E 01	0.0000000000000000E 00
	11	0.5150557500559767E-15	0.2500295714072456E 01	-0.1390006034002927E 01	0.0000000000000000E 00
	12	0.5150557500559767E-15	0.2790636600771119E 01	-0.1400926077311200E 01	0.0000000000000000E 00
	13	0.5150557500559767E-15	0.3000000000000000E 01	-0.1410607791653000E 01	0.0000000000000000E 00

Table 5.20 Samples at Time Step 250 of Surface Normal Components Array for M = 2, 12, and 22 and of Stress Subincrement Array for Layers K = 1 and 2

TIME STEP 250 [illegible] SURFACE NORMAL [illegible] COMPONENTS

M	N	XN(1,M,N)	XN(2,M,N)	XN(3,M,N)
2	1	[illegible]	[illegible]	[illegible]
	2	[illegible]	[illegible]	[illegible]
	3	[illegible]	[illegible]	[illegible]
	4	[illegible]	[illegible]	[illegible]
	[illegible]	[illegible]	[illegible]	[illegible]
	[illegible]	[illegible]	[illegible]	[illegible]
	[illegible]	[illegible]	[illegible]	[illegible]
	[illegible]	[illegible]	[illegible]	[illegible]
	[illegible]	[illegible]	[illegible]	[illegible]
	[illegible]	[illegible]	[illegible]	[illegible]
	[illegible]	[illegible]	[illegible]	[illegible]
	[illegible]	[illegible]	[illegible]	[illegible]
	[illegible]	[illegible]	[illegible]	[illegible]
12	[illegible]	[illegible]	[illegible]	[illegible]
	[illegible]	[illegible]	[illegible]	[illegible]
	[illegible]	[illegible]	[illegible]	[illegible]
	[illegible]	[illegible]	[illegible]	[illegible]
	[illegible]	[illegible]	[illegible]	[illegible]
	[illegible]	[illegible]	[illegible]	[illegible]
	[illegible]	[illegible]	[illegible]	[illegible]
	[illegible]	[illegible]	[illegible]	[illegible]
	[illegible]	[illegible]	[illegible]	[illegible]
	[illegible]	[illegible]	[illegible]	[illegible]
	[illegible]	[illegible]	[illegible]	[illegible]
	[illegible]	[illegible]	[illegible]	[illegible]
	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
	[illegible]	[illegible]	[illegible]	[illegible]
	[illegible]	[illegible]	[illegible]	[illegible]
	[illegible]	[illegible]	[illegible]	[illegible]
	[illegible]	[illegible]	[illegible]	[illegible]
	[illegible]	[illegible]	[illegible]	[illegible]
	[illegible]	[illegible]	[illegible]	[illegible]
	[illegible]	[illegible]	[illegible]	[illegible]
	[illegible]	[illegible]	[illegible]	[illegible]
	[illegible]	[illegible]	[illegible]	[illegible]
	[illegible]	[illegible]	[illegible]	[illegible]
	[illegible]	[illegible]	[illegible]	[illegible]
	[illegible]	[illegible]	[illegible]	[illegible]

TIME STEP [illegible] TIME [illegible] SUBDIVISIONS OF TIME INCREMENT IN STRESS

LAYER(K= 1)

[illegible]

LAYER(K= 2)

[illegible]

Table 5.21 Samples for N = 2, 12, and 22 of Displacement Increments Array During Time Increment 400 to 500 and of Cartesian Coordinates and Pressure Arrays at Time Step 500

[illegible]

N	M	UX(N,M)	UY(N,M)	UZ(N,M)
2	1	[illegible]	[illegible]	[illegible]
	2	[illegible]	[illegible]	[illegible]
	3	[illegible]	[illegible]	[illegible]
	4	[illegible]	[illegible]	[illegible]
	5	[illegible]	[illegible]	[illegible]
	6	[illegible]	[illegible]	[illegible]
	7	[illegible]	[illegible]	[illegible]
	8	[illegible]	[illegible]	[illegible]
	9	[illegible]	[illegible]	[illegible]
	10	[illegible]	[illegible]	[illegible]
	11	[illegible]	[illegible]	[illegible]
	12	[illegible]	[illegible]	[illegible]
	13	[illegible]	[illegible]	[illegible]
12	1	[illegible]	[illegible]	[illegible]
	2	[illegible]	[illegible]	[illegible]
	3	[illegible]	[illegible]	[illegible]
	4	[illegible]	[illegible]	[illegible]
	5	[illegible]	[illegible]	[illegible]
	6	[illegible]	[illegible]	[illegible]
	7	[illegible]	[illegible]	[illegible]
	8	[illegible]	[illegible]	[illegible]
	9	[illegible]	[illegible]	[illegible]
	10	[illegible]	[illegible]	[illegible]
	11	[illegible]	[illegible]	[illegible]
	12	[illegible]	[illegible]	[illegible]
	13	[illegible]	[illegible]	[illegible]
22	1	[illegible]	[illegible]	[illegible]
	2	[illegible]	[illegible]	[illegible]
	3	[illegible]	[illegible]	[illegible]
	4	[illegible]	[illegible]	[illegible]
	5	[illegible]	[illegible]	[illegible]
	6	[illegible]	[illegible]	[illegible]
	7	[illegible]	[illegible]	[illegible]
	8	[illegible]	[illegible]	[illegible]
	9	[illegible]	[illegible]	[illegible]
	10	[illegible]	[illegible]	[illegible]
	11	[illegible]	[illegible]	[illegible]
	12	[illegible]	[illegible]	[illegible]
	13	[illegible]	[illegible]	[illegible]

[illegible]

N	M	X(N,M, 500)	Y(N,M, 500)	Z(N,M, 500)	PRESSURE P(N,M)
2	1	[illegible]	[illegible]	[illegible]	[illegible]
	2	[illegible]	[illegible]	[illegible]	[illegible]
	3	[illegible]	[illegible]	[illegible]	[illegible]
	4	[illegible]	[illegible]	[illegible]	[illegible]
	5	[illegible]	[illegible]	[illegible]	[illegible]
	6	[illegible]	[illegible]	[illegible]	[illegible]
	7	[illegible]	[illegible]	[illegible]	[illegible]
	8	[illegible]	[illegible]	[illegible]	[illegible]
	9	[illegible]	[illegible]	[illegible]	[illegible]
	10	[illegible]	[illegible]	[illegible]	[illegible]
	11	[illegible]	[illegible]	[illegible]	[illegible]
	12	[illegible]	[illegible]	[illegible]	[illegible]
	13	[illegible]	[illegible]	[illegible]	[illegible]
12	1	[illegible]	[illegible]	[illegible]	[illegible]
	2	[illegible]	[illegible]	[illegible]	[illegible]
	3	[illegible]	[illegible]	[illegible]	[illegible]
	4	[illegible]	[illegible]	[illegible]	[illegible]
	5	[illegible]	[illegible]	[illegible]	[illegible]
	6	[illegible]	[illegible]	[illegible]	[illegible]
	7	[illegible]	[illegible]	[illegible]	[illegible]
	8	[illegible]	[illegible]	[illegible]	[illegible]
	9	[illegible]	[illegible]	[illegible]	[illegible]
	10	[illegible]	[illegible]	[illegible]	[illegible]
	11	[illegible]	[illegible]	[illegible]	[illegible]
	12	[illegible]	[illegible]	[illegible]	[illegible]
	13	[illegible]	[illegible]	[illegible]	[illegible]
22	1	[illegible]	[illegible]	[illegible]	[illegible]
	2	[illegible]	[illegible]	[illegible]	[illegible]
	3	[illegible]	[illegible]	[illegible]	[illegible]
	4	[illegible]	[illegible]	[illegible]	[illegible]
	5	[illegible]	[illegible]	[illegible]	[illegible]
	6	[illegible]	[illegible]	[illegible]	[illegible]
	7	[illegible]	[illegible]	[illegible]	[illegible]
	8	[illegible]	[illegible]	[illegible]	[illegible]
	9	[illegible]	[illegible]	[illegible]	[illegible]
	10	[illegible]	[illegible]	[illegible]	[illegible]
	11	[illegible]	[illegible]	[illegible]	[illegible]
	12	[illegible]	[illegible]	[illegible]	[illegible]
	13	[illegible]	[illegible]	[illegible]	[illegible]

Table 5.22 Samples at Time Step 500 of Surface Normal Components Array for M = 2, 12, and 22 and of Stress Subincrement Array for Layers K = 1 and 2

TIME STEP 500 TIME [illegible]

SURFACE NORMAL VECTOR COMPONENTS

M	N	SN1(M,N)	SN2(M,N)	SN3(M,N)
2	1	[illegible]	[illegible]	[illegible]
	2	[illegible]	[illegible]	[illegible]
	3	[illegible]	[illegible]	[illegible]
	4	[illegible]	[illegible]	[illegible]
	5	[illegible]	[illegible]	[illegible]
	6	[illegible]	[illegible]	[illegible]
	7	[illegible]	[illegible]	[illegible]
	8	[illegible]	[illegible]	[illegible]
	9	[illegible]	[illegible]	[illegible]
	10	[illegible]	[illegible]	[illegible]
	11	[illegible]	[illegible]	[illegible]
	12	[illegible]	[illegible]	[illegible]
	13	[illegible]	[illegible]	[illegible]
12	1	[illegible]	[illegible]	[illegible]
	2	[illegible]	[illegible]	[illegible]
	3	[illegible]	[illegible]	[illegible]
	4	[illegible]	[illegible]	[illegible]
	5	[illegible]	[illegible]	[illegible]
	6	[illegible]	[illegible]	[illegible]
	7	[illegible]	[illegible]	[illegible]
	8	[illegible]	[illegible]	[illegible]
	9	[illegible]	[illegible]	[illegible]
	10	[illegible]	[illegible]	[illegible]
	11	[illegible]	[illegible]	[illegible]
	12	[illegible]	[illegible]	[illegible]
	13	[illegible]	[illegible]	[illegible]
22	1	[illegible]	[illegible]	[illegible]
	2	[illegible]	[illegible]	[illegible]
	3	[illegible]	[illegible]	[illegible]
	4	[illegible]	[illegible]	[illegible]
	5	[illegible]	[illegible]	[illegible]
	6	[illegible]	[illegible]	[illegible]
	7	[illegible]	[illegible]	[illegible]
	8	[illegible]	[illegible]	[illegible]
	9	[illegible]	[illegible]	[illegible]
	10	[illegible]	[illegible]	[illegible]
	11	[illegible]	[illegible]	[illegible]
	12	[illegible]	[illegible]	[illegible]
	13	[illegible]	[illegible]	[illegible]

TIME STEP 500 TIME [illegible] SUBDIVISIONS OF TIME INCREMENT IN STRESS

[illegible](M,N, 1)

N	M	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22
1		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
5		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
6		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
7		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
9		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
11		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
12		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
13		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

[illegible](M,N, 2)

N	M	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22
1		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
5		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
6		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
7		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
9		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
11		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
12		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
13		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Table 5.23 Samples of Energy Balance and of Surface Strains Printed Output at Time Steps 250 and 500

TIME STEP 250 TIME= 0.37500000E-03

SURFACE STRAINS / STRAIN GAGE READING

ETA1	ETA2	M	N	FACE	ANGLE 0	ANGLE 90	ANGLE		ANGLE	
.000	3.000	2.000	13.000	INNER	-0.21843296E 00	0.18997300E-01	45.00	-0.91924415E-01	135.00	-0.91924415E-01
.000	3.000	2.000	13.000	OUTER	-0.13090572E 00	0.19724857E-01	45.00	-0.52592050E-01	135.00	-0.52592050E-01
1.162	3.000	7.000	13.000	INNER	-0.10754889E 00	0.39073041E-01	45.00	-0.31459405E-01	135.00	-0.31459405E-01
1.162	3.000	7.000	13.000	OUTER	-0.11384974E 00	0.28376254E-01	45.00	-0.40098962E-01	135.00	-0.40098932E-01
2.323	1.500	12.000	7.000	INNER	-0.13332724E-01	0.11780287E-01	45.00	0.67773029E-02	135.00	-0.82282890E-02
2.323	1.500	12.000	7.000	OUTER	0.72246290E-02	0.56118482E-02	45.00	0.37141386E-02	135.00	0.91157369E-02

TIME STEP 250 TIME= 0.37500000E-03 KINETIC= 0.48234038E 02 ELASTIC= 0.51034366E 02 PLASTIC= 0.71923544E 04
TOTAL ENERGY= 0.72916228E 04

TIME STEP 500 TIME= 0.61997784E-03

SURFACE STRAINS / STRAIN GAGE READING

ETA1	ETA2	M	N	FACE	ANGLE 0	ANGLE 90	ANGLE		ANGLE	
.000	3.000	2.000	13.000	INNER	-0.21902752E 00	0.19025327E-01	45.00	-0.92161030E-01	135.00	-0.92167946E-01
.000	3.000	2.000	13.000	OUTER	-0.13087125E 00	0.19016056E-01	45.00	-0.52960856E-01	135.00	-0.52954403E-01
1.162	3.000	7.000	13.000	INNER	-0.11230353E 00	0.37450252E-01	45.00	-0.34533496E-01	135.00	-0.34534025E-01
1.162	3.000	7.000	13.000	OUTER	-0.10930036E 00	0.29364833E-01	45.00	-0.37480294E-01	135.00	-0.37454609E-01
2.323	1.500	12.000	7.000	INNER	-0.13365566E-01	0.12458008E-01	45.00	0.66152659E-02	135.00	-0.74052023E-02
2.323	1.500	12.000	7.000	OUTER	0.70857540E-02	0.64973669E-02	45.00	0.38840209E-02	135.00	0.96908131E-02

TIME STEP 500 TIME= 0.61997784E-03 KINETIC= 0.11639291E 00 ELASTIC= 0.31063251E 02 PLASTIC= 0.72053024E 04
TOTAL ENERGY= 0.72916228E 04

TAPE 1 WRITTEN, NCYCLE= 500 TIME= 0.61997784E-03

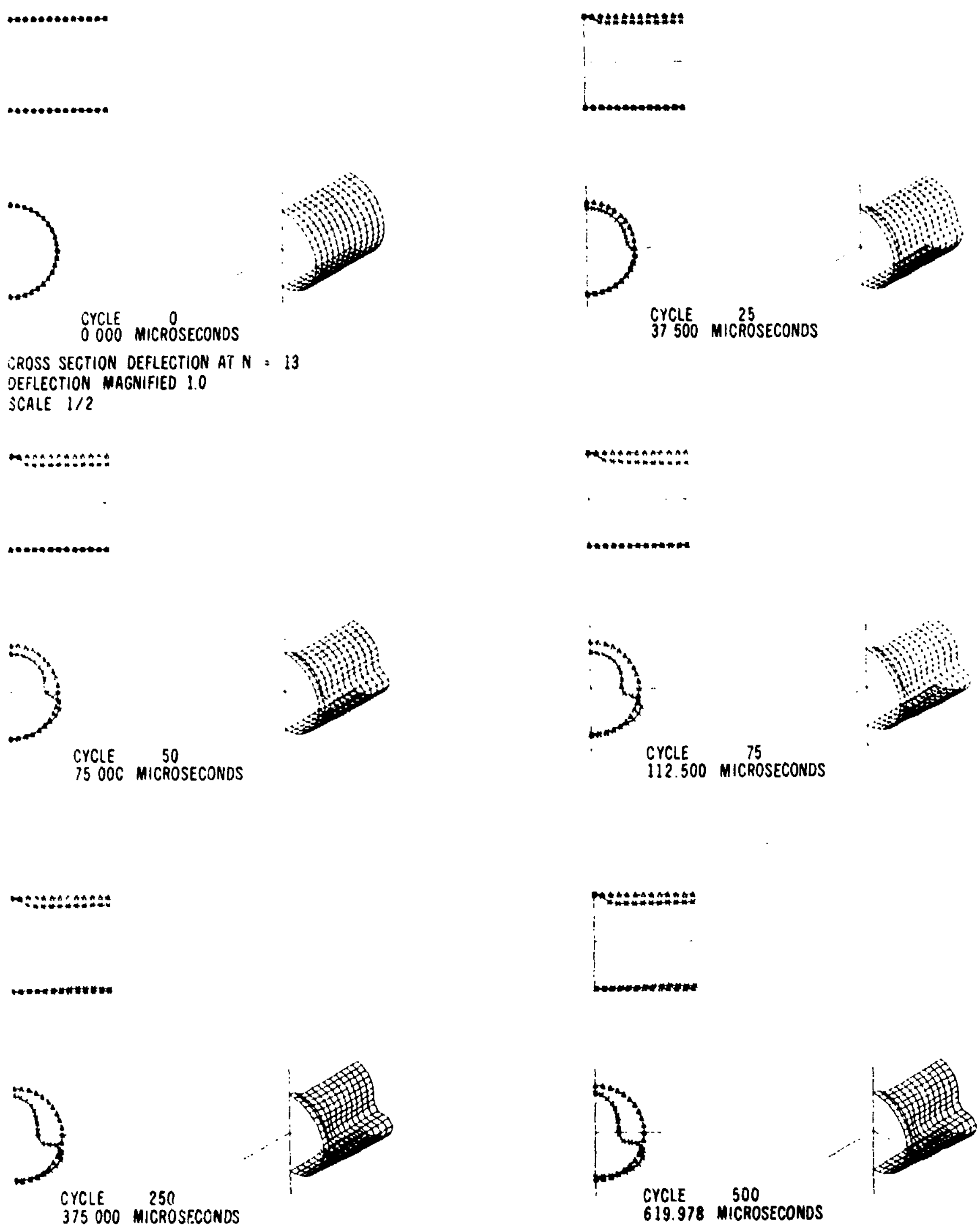
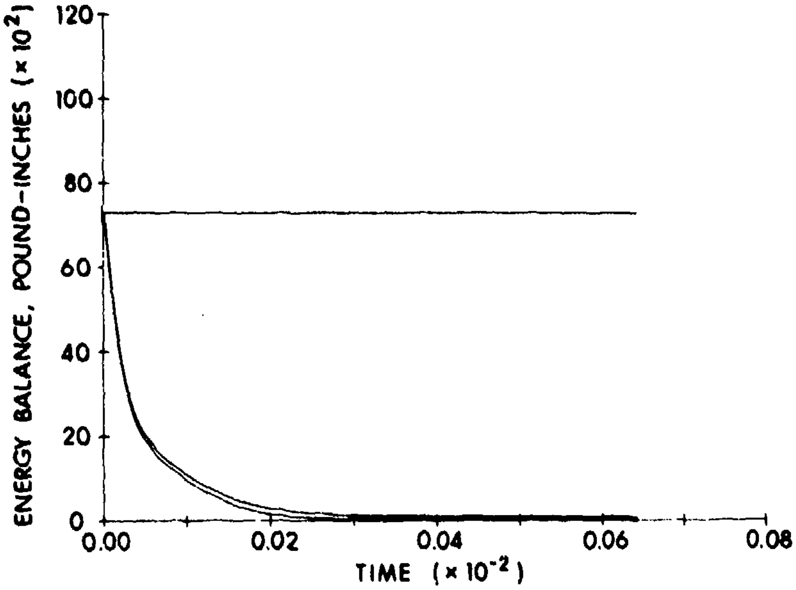
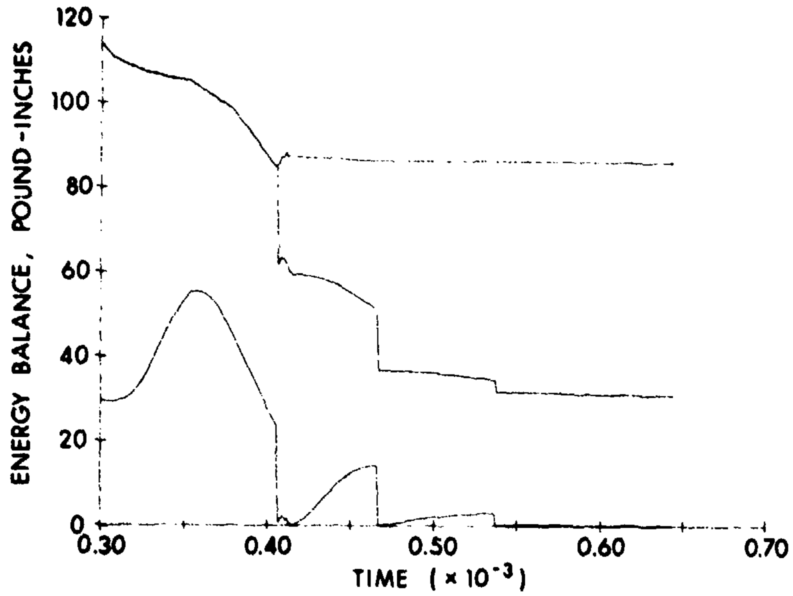


Figure 5.8 Isometric and Cross-sectional Cal Comp Plots of the Deforming Middle Surface at Selected Time Steps Showing True (Unmagnified) Displacements



a. Entire history of the energy balance from 0 to 643 microseconds, showing in ascending order the kinetic energy, total energy, total energy plus damping work, and external work.



b. Blow up of the history of the energy balance from 300 to 643 microsecond, showing in ascending order the kinetic energy, total energy, and total energy plus damping work.

Figure 5.9 Cal Comp Plots of the Energy Balance for the Impulsively Loaded Cylinder

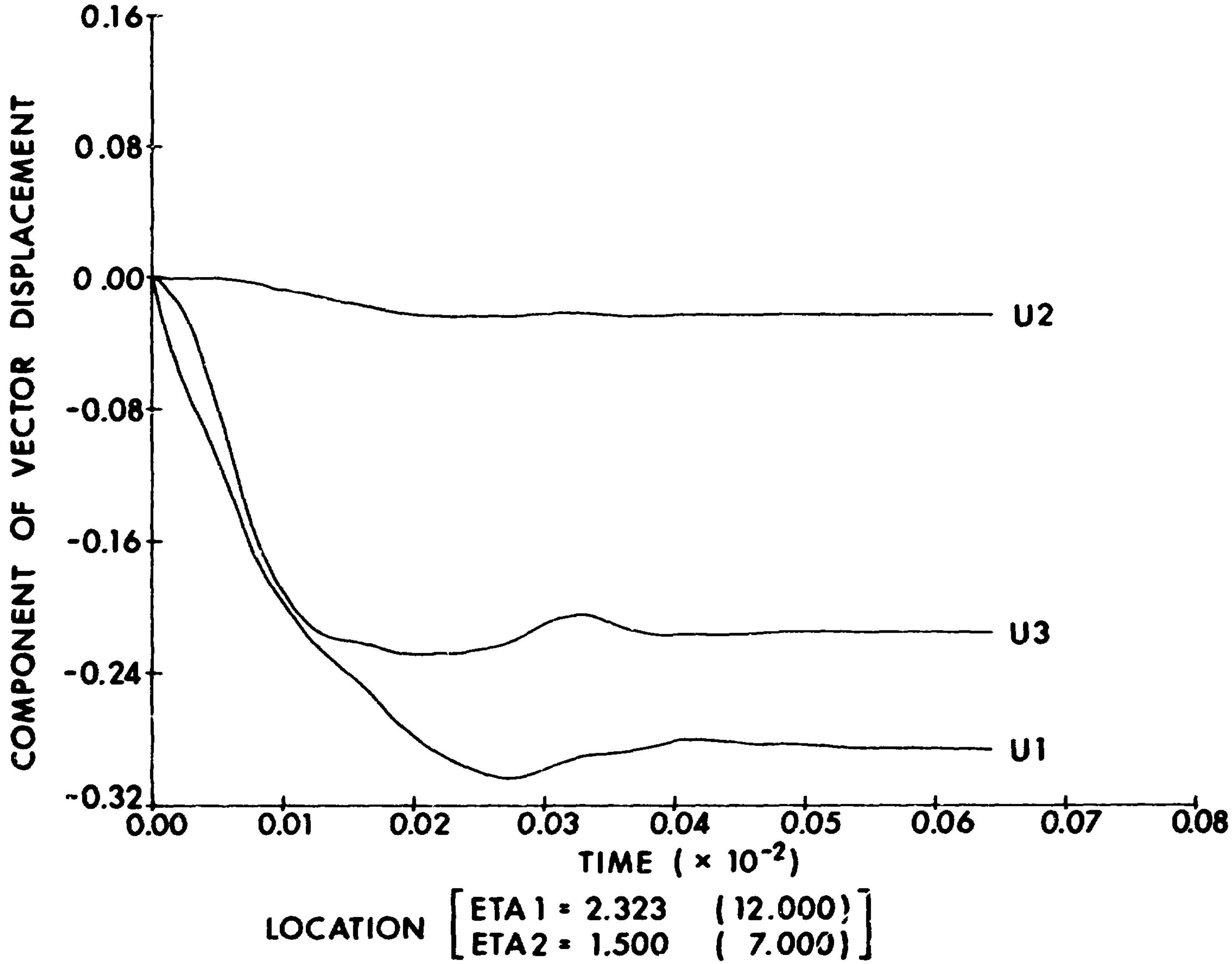
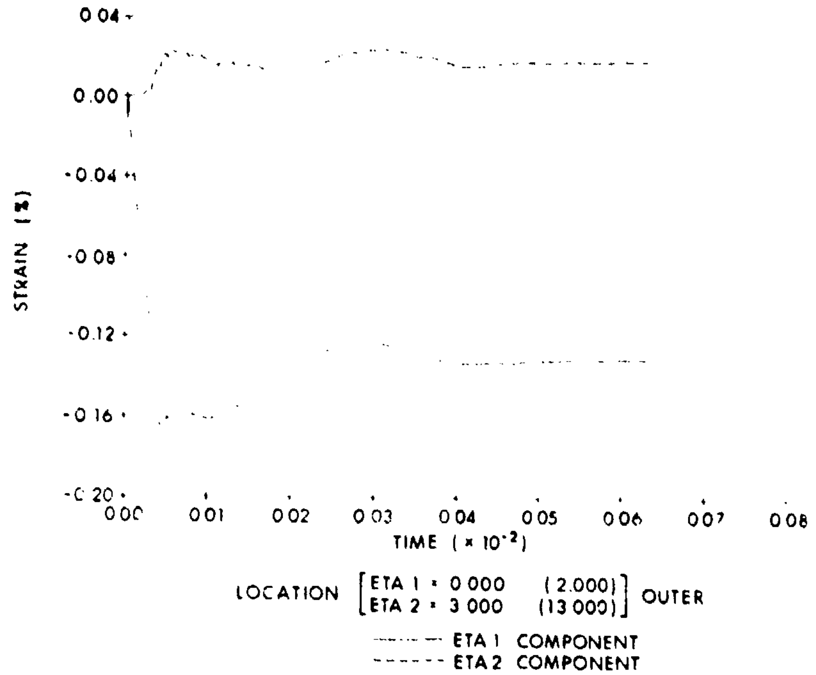
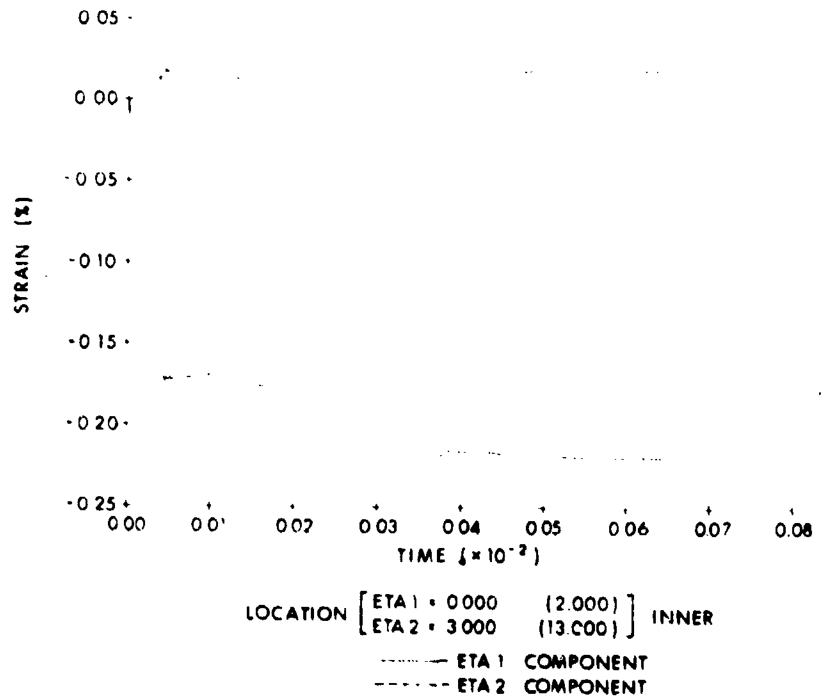


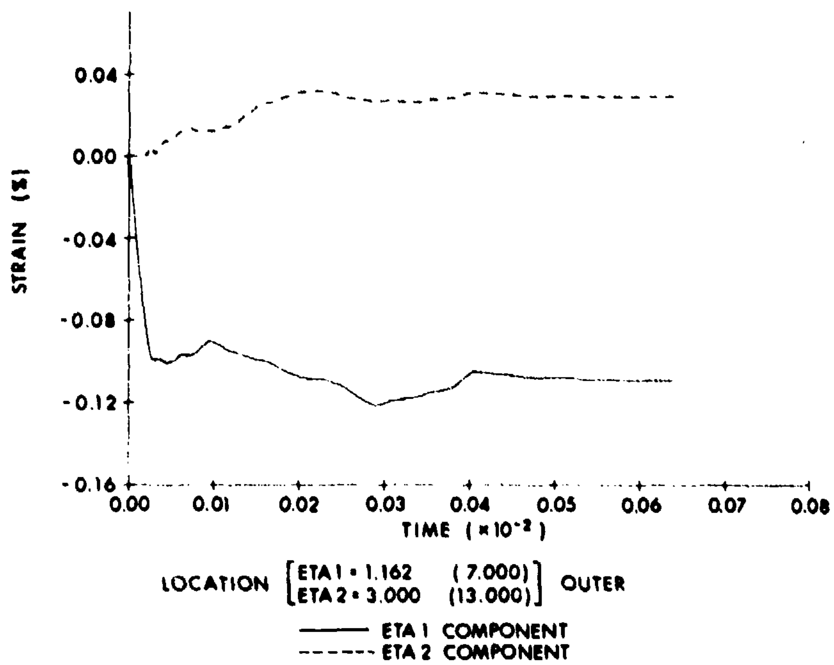
Figure 5.10 Cal Comp Plot of the History of the Deflection at Point 90° from Crown Line and Midway Between Clamped and Symmetry Edges

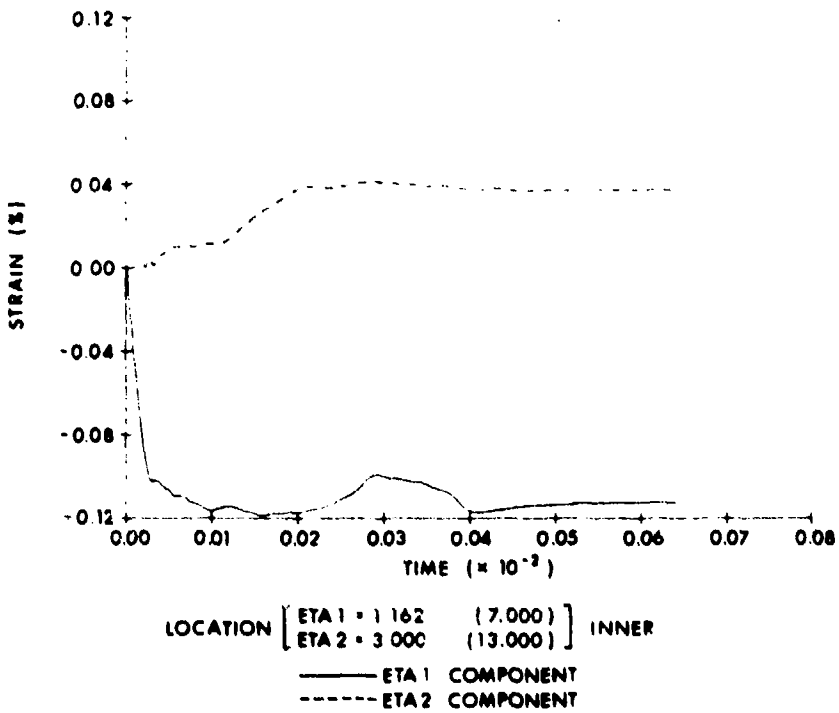




a. Strains on the crown line at half the axial length.

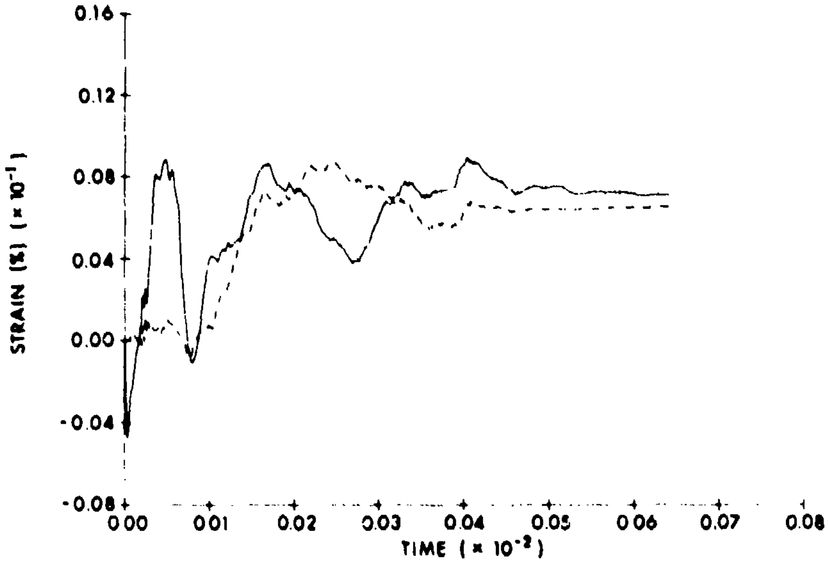
Figure 5.11 CalComp Plots of the History of the Surface Elongational Strains Along the Mesh Directions

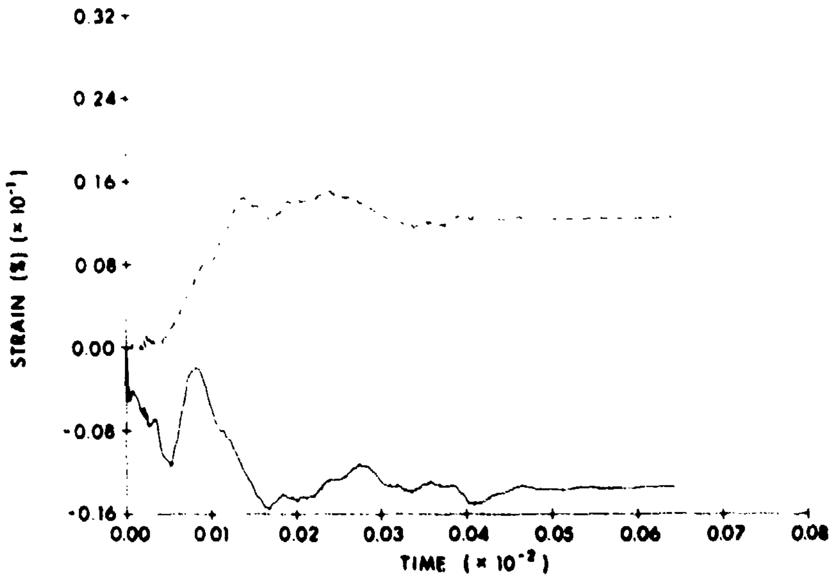




b. Strains at 45° from the crown line and half the axial length.

Figure 5.11 (Continued)





c. Strains at 90° from the crown line and quarter the axial length.

Figure 5.11 (Continued)

6. IMPLEMENTATION OF LOADINGS AND INITIAL GEOMETRIES

REPSIL can be made to accept arbitrary impulse loadings, arbitrary time and space varying pressure loadings, and, within certain limits, arbitrary initial shell geometries. This involves a small amount of programming confined to certain REPSIL subroutine: INVEL, PRESS and INGEOM. This chapter describes the acceptable ways of implementing loadings and geometries into REPSIL so as to be compatible with the code's formulation.

6.1 Two Approaches to Implementing Loads and Geometries

Implementing a loading or an initial geometry in REPSIL basically involves assigning to each mesh point (m,n) a value of initial impulse velocity or a value of pressure for each time step, or values of the initial coordinates of the middle surface. These values can be assigned in two ways: first, they can be read off a tape or cards and assigned to the mesh points by the appropriate subroutine; second, an analytical expression programmed into the appropriate subroutine can generate these values, which the subroutine then assigns to the mesh points. The first approach requires that data be generated outside the program and be read onto a tape or punched onto cards. The data itself can be generated analytically, numerically or experimentally -- this approach will accept all; therein lies its advantage, especially for *ad hoc* problems. Its drawbacks are the tediousness of reading onto a tape or punching onto cards large numbers of values especially when many mesh points are employed, and necessity of regenerating new data for a given physical problem whenever the number of mesh intervals is changed. The second approach has none of these drawbacks, but does require that the data be expressible analytically. These analytical expressions are most conveniently written as functions of the underlying material coordinates (η^1, η^2) rather than the mesh numbers (m,n), for then the forms of the expressions are not affected by a change in the number of mesh intervals. The value of such an expression at a mesh point (m,n) is obtained as the value of the function at the material coordinates (η^1, η^2) corresponding to the point (m,n) :

$$\eta^1(m) = \eta^1_o + (m-1)\ \Delta\eta^1 \quad , \quad \eta^2(n) = \eta^2_o + (n-1)\ \Delta\eta^2 \quad * \qquad (6.1)$$

* These equations are closer to the indexing of m and n used in the program than (2.1), reflecting the fact that FORTRAN does not permit zero values of array indices.

where, for M x N mesh intervals, $1 \leq m \leq M + 1$, $1 \leq n \leq N + 1$ and

$$\Delta\eta^1 = \frac{\eta_f^1 - \eta_o^1}{M} \quad , \quad \Delta\eta^2 = \frac{\eta_f^2 - \eta_o^2}{N} \quad . \tag{6.2}$$

In the subsequent sections, in which the implementation of the loading and geometry subroutines are individually described, the two approaches are again covered, as they apply to each subroutine.

6.2 Subroutine INVEL

Initial impulse loadings are specified in subroutine INVEL. At the zeroth time step INVEL assigns a value of impulse velocity to each mesh point. The velocity is free to vary spatially over the shell surface, but at present is restricted to be directed along the normal - a minor restriction that is easily remedied by allowing tangential components to be defined through the middle surface basis vectors, see Section 2.3.

As stated in the previous section, two approaches can be used in programming INVEL. With the first approach, INVEL simply reads off cards or a tape a sequence of values corresponding to the normal velocity v at each mesh point. Each value is immediately multiplied by the surface normal at the mesh point to give components of the velocity

$$v^i = v\, n^i \ , \tag{6.3}$$

which are then stored in the as yet unused three M x N arrays $\Delta u^i(m,n)$. A form of this approach is used for the subroutine INVEL presently in REPSIL, which is specified on input Card 15 and 16, see Section 3.2. (A listing of this subroutine is given in Appendix E.)

The second approach requires some analytic expressions of the form

$$v = v\,(\eta^1, \eta^2) \tag{6.4}$$

for the normal velocity as a function of the material coordinates. The material coordinates are limited to the rectangular domain specified in INGEOM, see (6.10) of Section 6.4. With this approach, (6.1) and (6.4) are both programmed into INVEL; (6.2) need not be included since the increments $\Delta\eta^\alpha$ are calculated in INGEOM. INVEL determines the normal velocity at mesh points by calculating the values of the material

coordinates at the mesh point (m,n) using (6.1) and substituting these values in (6.4) to obtain

$$v(m,n) = v(\eta^1(m),\ \eta^2(n)). \tag{6.5}$$

As with the first approach, these values are immediately multiplied by the normal to obtain the components of the velocity at (m,n), which are then stored in the three M x N arrays $\Delta u^i(m,n)$.

6.3 Subroutine PRESS

Time/space varying pressure loads are specified in subroutine PRESS. Each time step this subroutine assigns a value of pressure to each mesh point.

With the first of the two approaches to programming already mentioned, at each time step PRESS reads off cards or (more usually) a tape a sequence of values of the pressure P at each mesh point and stores these values in the M x N array P(m,n) for later use in subroutine MOTION, see Section 2.3. This approach has been used in REPSIL for a number of cases, in which the pressure data has been experimentally/numerically generated; reports on these cases are in preparation.

For the second approach, analytic expressions for the pressure as a function of the material coordinates and the time:

$$P = P\,(\,\eta^1,\ \eta^2,\ t) \tag{6.6}$$

are programmed into PRESS, along with (6.1); the material coordinates are limited to the rectangular domain (6.10) specified in INGEOM. At each time step ℓ, PRESS calculates the pressure P at each mesh point (m,n) by determining the material coordinates of the mesh point from (6.1), the value of time at time step ℓ from

$$t(\ell) = \ell\ \Delta t \tag{6.7}$$

and substituting these values in (6.6):

$$P(m,n) = P(\eta^1(m),\ \eta^2(n),\ t(\ell)). \tag{6.8}$$

As with the first approach, these values are stored in the M x N array

P (m,n). This approach is used to generate the pressure data for example problem 1 (Section 5.1); a listing of this PRESS subroutine is in Appendix E.

6.4 Subroutine INGEOM

The initial geometry of the shell is specified in subroutine INGEOM. A new subroutine INGEOM must be written for each initial geometry or at least each family of initial geometries.* The only restrictions on the admissable geometries are that the middle surface be simply connected and bounded by four smooth edges, such that none of the corners formed by intersecting edges are reentrant or straight.

Particularizing the remarks of Section 6.1 to initial geometries, the basic function of INGEOM is to set up a correspondence between mesh numbers (m,n) and coordinates y^i in 3-space through which the middle surface of the shell initially passes; this correspondence should be one-to-one. As already mentioned, two approaches can be used in programming INGEOM. First, INGEOM can just comprise instructions for reading off cards or a tape a sequence of ordered triplexes (y^1, y^2, y^3) and storing these in three M x N arrays $y^1(m,n)$, $y^2(m,n)$ and $y^3(m,n)$. Of course, the sequence of triplexes y^i must be chosen carefully, not only for sake of obtaining a one-to-one correspondence, but also in order that the correspondence be topologically continuous in the following sense: neighboring points in space are assigned adjacent mesh numbers in the proper order, as indicated in Figure 6.1. This approach requires that the increments in material coordinates, $\Delta\eta^1$ and $\Delta\eta^2$, be assigned convenient values that are subsequently used to form finite difference quotients. Also, it is important that the number of mesh intervals M and N assigned elsewhere in the program jibe with the spacing implied by subroutine INGEOM.

The second approach uses one or more sets of analytic expressions giving the Cartesian coordinates y^i of the middle surface parametrically as functions of the material coordinates η^α:

$$y^1 = y^1(\eta^1,\eta^2) \ , \ y^2 = y^2(\eta^1,\eta^2) \ , \ y^3 = y^3(\eta^1,\eta^2) \ . \qquad (6.9)$$

These functions are required to be continuous and one-to-one in the domain over which the material coordinates vary; the domain itself is

* REPSIL has INGEOM subroutines for a flat plate, cylindrical shell and conical shell programmed, see Section 3.2 and Appendix E.

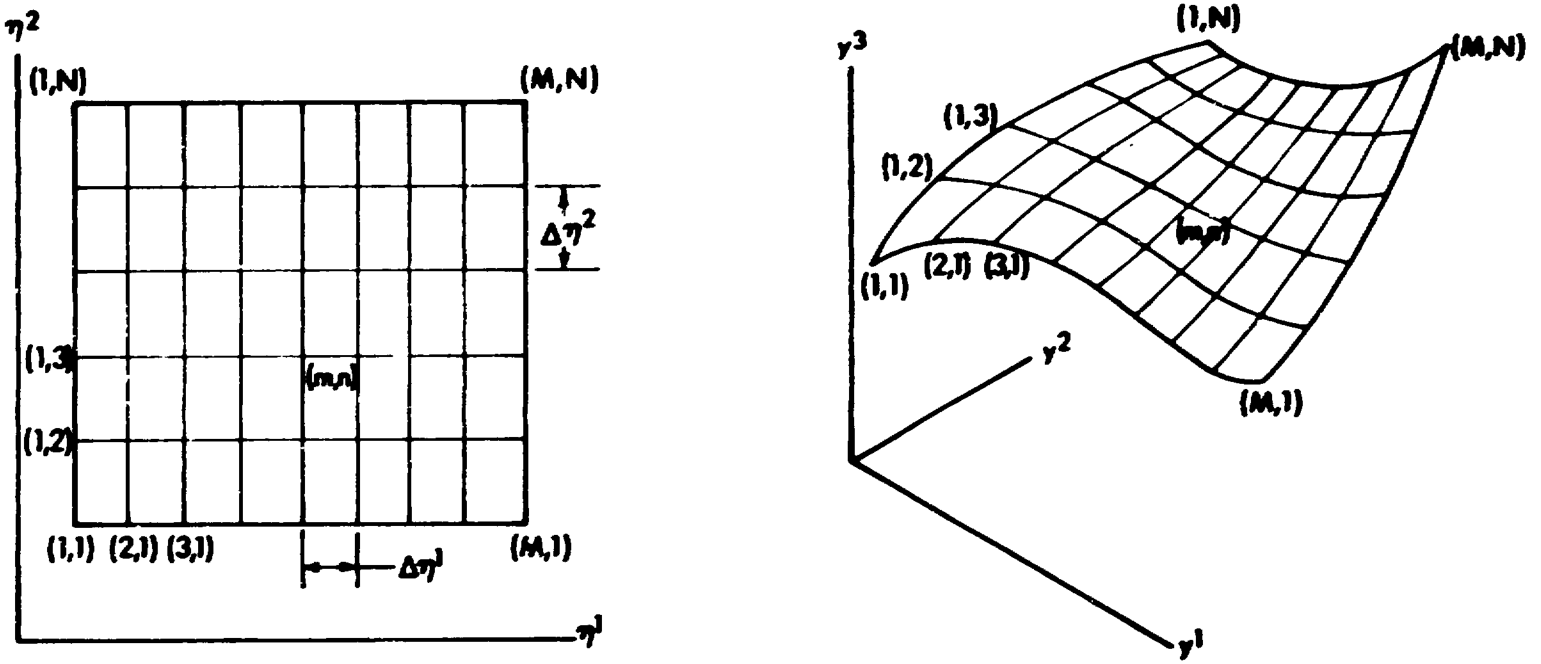


Figure 6.1 Mapping of Material Coordinate Mesh onto the Middle Surface of Shell

limited to some rectangle in the material coordinates plane:

$$\eta_o^1 \leq \eta^1 \leq \eta_f^1 \quad , \quad \eta_o^2 \leq \eta^2 \leq \eta_f^2 \quad . \tag{6.10}$$

Consequently, the parametric representation of the middle surface is simply a one-to-one continuous map of a rectangle in the η^1, η^2, plane into 3-space, from which automatically follow the aforementioned topological continuity and restrictions on the boundary. It should be noted that the parametric representation is not unique--many exist for a given surface. Moreover, the material coordinates need not have physical significance, such as arc length, angle, etc., although often a simple transformation can give them such meaning.

With the second approach, equations (6.9) are programmed into INGEOM, as well as (6.1) and (6.2). For a given physical problem, the extent of the shell is fixed by the limits on the material coordinates (6.10). Once the number of mesh intervals M x N are specified, the subroutine computes the constant intervals $\Delta\eta^\alpha$ using (6.2) and the values of the material coordinates $\eta^1(m)$ and $\eta^2(n)$ at the mesh points (m,n) using (6.1). Substituting these values of the material coordinates into the analytical expression (6.9), the subroutine calculates the Cartesian coordinates of the mesh points:

$$y^1 = y^1(m,n) \quad , \quad y^2 = y^2(m,n) \quad , \quad y^3 = y^3(m,n) \quad , \tag{6.11}$$

thus mapping the rectangular M x N mesh in the η^1, η^2 plane into a curvelinear mesh in 3-space, as pictured in Figure 6.1. As with the first approach, the subroutine stores these values in three M x N arrays.

The second approach to programming INGEOM will be illustrated with an example of a shell having for its initial middle surface a frustum of a circular cone with an axial length L, a small radius R_o and large radius R_f. Let the frustrum be located relative to the y^1, y^2, y^3 axis as shown in Figure 3.8. With this orientation the coordinates of the frustum are parametrically given by

$$y^1 = \eta^1 \sin \eta^2 \ , \ y^2 = \frac{\eta^1 - R_o}{R_f - R_o} L, \ y^3 = \eta^1 \cos \eta^2 \ , \tag{6.12}$$

with the material coordinates limited to the domain

$$R_o \le \eta^1 \le R_f \;\&\; 0 \le \eta^2 < 2\pi \quad . \tag{6.13}$$

In this representation both parameters have physical significance: η^1 being the radius from the cone axis and η^2 the angle about the axis.

This representation is not unique; for example, η^1 can be replaced by a parameter $\hat{\eta}^1$ measuring arclength along generators through the transformation

$$\hat{\eta}^1 = \eta^1 \csc \alpha \quad ,$$

with α the cone angle:

$$\alpha = \arctan \frac{R_f - R_o}{L} \quad , \tag{6.14}$$

resulting in the representation

$$y^1 = \hat{\eta}^1 \sin\alpha \;\; \sin\eta^2 \; , \; y^2 = (\hat{\eta}^1 - S_o) \cos\alpha \; , \; y^3 = \hat{\eta}^1 \sin\alpha \cos\eta^2 \; , \tag{6.15}$$

with domain

$$S_o\ (= R_o \csc\alpha) \le \hat{\eta}^1 \le S_f\ (= R_f \csc\alpha),\ 0 \le \eta^2 < 2\pi \tag{6.16}$$

Clearly, a transformation replacing η^2 by circumferential arclength cannot exist, for it is impossible for the same angle η^2 at different location along the cone axis to subtend equal arclength on the surface of a cone. For the same reason, the image of an M x N mesh under the representation (6.15) or for that matter (6.12) will yield a curvilinear rectangular mesh with nonequal rectangles: as shown in Figures 3.8 and 6.2a, while the meridional lengths of rectangles are equal, the circumferential lengths increase with the radius. However, it is possible through a judicious transformation of η^1 or $\hat{\eta}^1$ into say $\bar{\eta}^1$ to obtain nonuniform meridional increments that give similar rectangles (i.e. rectangles of constant side ratio). Using differentials, the condition that constant increments $\Delta\bar{\eta}^1$ and $\Delta\eta^2$ give rectangles with a constant meridional to circumferential side ratio of κ can be written as

$$\left| \frac{\partial y^i}{\partial \bar{\eta}^1} \Delta \bar{\eta}^1 \right| \Big/ \left| \frac{\partial y^i}{\partial \eta^2} \Delta \eta^2 \right| = \kappa$$

Assuming that $\hat{\eta}^1$ is a function of $\bar{\eta}^1$ and using (6.15), this condition is shown to be equivalent to the differential equation

$$\frac{d\hat{\eta}^1}{d\bar{\eta}^1} = K \sin\alpha \ \hat{\eta}^1 \quad , \quad K = \kappa \frac{\Delta \eta^2}{\Delta \bar{\eta}^1} \quad .$$

Chosing the lower limit on $\bar{\eta}^1$ to be zero for the sake of convenience, the solution of this differential equation yield the transformation

$$\hat{\eta}^1 = S_o \, e^{K \sin\alpha \ \bar{\eta}^1} \quad ,$$

in the range $0 \leq K \sin \alpha \ \bar{\eta}^1 \leq \ln \frac{S_f}{S_o}$. This transformation is substituted in (6.15) to give the parametric representation

$$y^1 = S_o \sin\alpha \times \sin\eta^2 \, e^{K \sin\alpha \ \bar{\eta}^1}$$

$$y^2 = S_o \cos\alpha \times (e^{K \sin\alpha \ \bar{\eta}^1} - 1) \tag{6.17}$$

$$y^3 = S_o \sin\alpha \times \cos\eta^2 \, e^{K \sin\alpha \ \bar{\eta}^1}$$

over the domain

$$0 \leq K \sin\alpha \ \bar{\eta}^1 \leq \ln \frac{S_f}{S_o} \quad , \quad 0 \leq \eta^2 < 2\pi \quad . \tag{6.18}$$

Since the program specified the number of mesh intervals M and N rather than the increments $\Delta\eta^\alpha$, the constant K can be conveniently set equal to unity. Also (6.15) and (6.17) can be summarized in a single general representation by a change in the scale of $\hat{\eta}^1$, giving the representation

$$y^1 = S_o \sin\alpha \ \sin\eta^2 \, f(\eta^1 \sin\alpha)$$

$$y^2 = S_o \cos\alpha \ [f(\eta^1 \sin\alpha) - 1] \tag{6.19}$$

$$y^3 = S_o \sin\alpha \ \cos\eta^2 \, f(\eta^1 \sin\alpha),$$

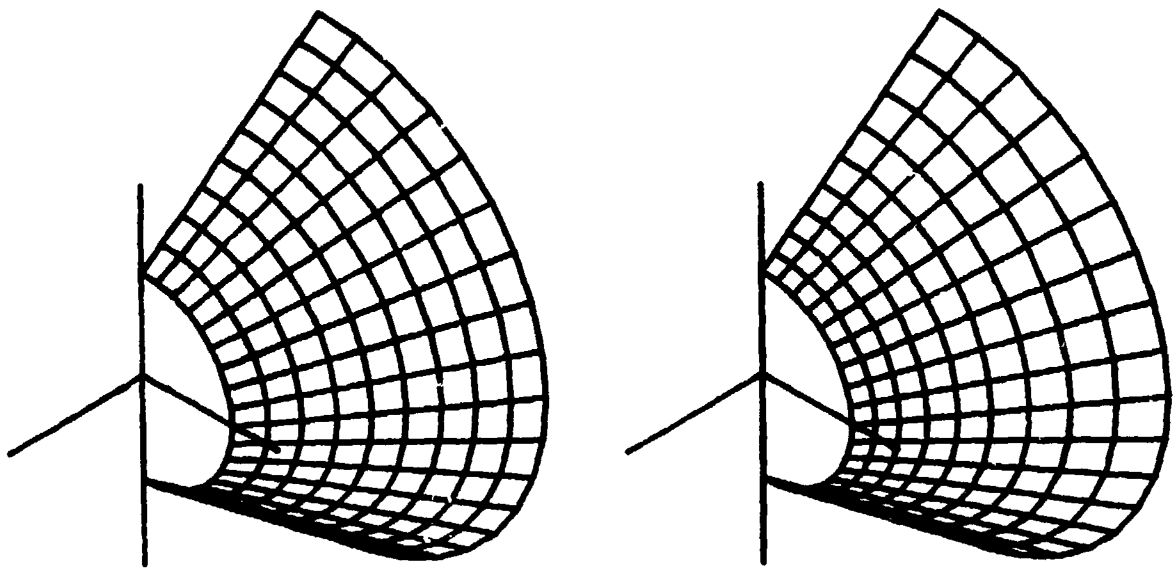
where for a uniform meridional spacing

$$f(\eta^1 \sin\alpha) = \eta^1 \sin\alpha \quad ; \quad 1 \le \eta^1 \sin\alpha \le \frac{S_f}{S_o} \tag{6.20}$$

and for a constant side ratio

$$f(\eta^1 \sin\alpha) = \exp(\eta^1 \sin\alpha) \quad ; \quad 0 \le \eta^1 \sin\alpha \le \ln \frac{S_f}{S_o} \quad . \tag{6.21}$$

The programming of this representation is straightforward and is included in the listing in Appendix E. Figure 6.2 shows the difference between the two types of spacing for given M x N mesh intervals, reproduced from Cal Comp plots.



a. Constant meridional increment b. Constant (almost square) side ratio

Figure 6.2 Comparison of the 18 x 9 Mesh Generated by Subroutine INGEOM for the Frustum of a Cone Using the Constant Meridional Increment Option and the Constant Mesh Proportions Option

REFERENCES

1. J.M. Santiago, "Formulation of the Large Deflection Shell Equations for Use in Finite Difference Structural Response Computer Codes", U.S. Army Ballistic Research Laboratories, Report No. 1571 February 1972.

2. L. Morino, J.W. Leech and E.A. Witmer, "PETROS 2: A New Finite-Difference Method and Program for the Calculation of Large Elastic-Plastic Dynamically-Induced Deformations of General Thin Shells", U.S. Army Ballistic Research Laboratories Contract Report No. 12, December 1969.

3. N.J. Huffington, Jr., "Numerical Analysis of Elastoplastic Stresses", U.S. Army Ballistic Research Laboratories, Memorandum Report No. 2006, September 1969.

4. R.E. Dahl, J.R. Beeler, Jr., and R.D. Bourquin, "GRAINS--A Quasi-Dynamic Code for Computer Simulation Experiments of Point and Line Defects in Metals", Battelle Northwest Laboratories, Report 1248, UC-32, January 1970.

APPENDIX A.
FINITE DIFFERENCE OPERATORS

The partial differential equations solved by the REPSIL code employ the following partial derivatives with respect to the material coordinates:

$$\frac{\partial}{\partial \eta^1} \; , \quad \frac{\partial}{\partial \eta^2} \; , \quad \frac{\partial^2}{(\partial \eta^1)^2} \; , \quad \frac{\partial^2}{\partial \eta^1 \partial \eta^2} \; , \quad \frac{\partial^2}{(\partial \eta^2)^2} \; .$$

These partials are approximated in the code by finite difference operators of order $| \Delta\eta |^2$ in accuracy. These operators, which were symbolized by their corresponding partials in Section 2.3, will now be given explicitly.

Let F(m,n) represent a typical mesh function (the position coordinates y^i or the bending resultants $M^{\alpha\beta}$, for example) defined over the domain of mesh points $m_i \leq m \leq m_f$ and $n_i \leq n \leq n_f$. At interior mesh points where the mesh numbers (m,n) satisfy $m_i < m < m_f$ and $n_i < n < n_f$, central difference operators, symbolized by the superscript c, are used:

$$\frac{\Delta F^c}{\Delta \eta^1}(m,n) \equiv \frac{F(m+1,n) - F(m-1,n)}{2\Delta\eta^1} \; ,$$

$$\frac{\Delta F^c}{\Delta \eta^2}(m,n) \equiv \frac{F(m,n+1) - F(m,n-1)}{2\Delta\eta^2} \; ,$$

$$\frac{\Delta^2 F^c}{(\Delta \eta^1)^2}(m,n) \equiv \frac{F(m+1,n) - 2F(m,n) + F(m-1,n)}{(\Delta\eta^1)^2} \; ,$$

$$\frac{\Delta^2 F^c}{(\Delta \eta^2)^2}(m,n) \equiv \frac{F(m,n+1) - 2F(m,n) + F(m,n-1)}{(\Delta\eta^2)^2} \; ,$$

$$\frac{\Delta^2 F^c}{\Delta \eta^1 \Delta\eta^2}(m,n) \equiv \frac{F(m+1,n+1) - F(m-1,n+1) - F(m+1,\, n-1) + F(m-1,n-1)}{\Delta\eta^1 \Delta\eta^2} \; ,$$

with $\Delta\eta^1$, $\Delta\eta^2$ (constant) increments in the material coordinates. Notice that the mixed partials operator involves the successive application of the first partial operators:

$$\frac{\Delta^2 F^c}{\Delta\eta 1 \Delta\eta 2}(m,n) = \frac{\frac{\Delta F^c}{\Delta\eta 2}(m+1,n) - \frac{\Delta F^c}{\Delta\eta 2}(m-1,n)}{2\Delta\eta 1} \equiv \frac{\frac{\Delta F^c}{\Delta\eta 1}(m,n+1) - \frac{\Delta F^c}{\Delta\eta 1}(m,n-1)}{2\Delta\eta 2}$$

or more compactly:

$$\frac{\Delta^2 F^c}{\Delta\eta 1 \Delta\eta 2} = \frac{\Delta}{\Delta\eta 1}\left(\frac{\Delta F^c}{\Delta\eta 2}\right)^c \equiv \frac{\Delta}{\Delta\eta 2}\left(\frac{\Delta F^c}{\Delta\eta 1}\right)^c$$

Along the boundaries where $m = m_i$ or m_f or $n = n_i$ or n_f central difference operators cannot be used to approximate all partials due to F(m,n) not being defined outside the domain of mesh points. Rather, some of the above central difference operators are replaced by either forward or backward difference operators. Specifically, along $m = m_i$ or m_f only the central difference operators $\frac{\Delta F^c}{\Delta\eta 2}$ and $\frac{\Delta^2 F^c}{(\Delta\eta 2)^2}$ are employed. The remaining operators are replaced by forward difference operators (denoted by the superscript f) along $m = m_i$:

$$\frac{\Delta F^f}{\Delta\eta 1}(m,n) = -\frac{3F(m,n) - 4F(m+1,n) + F(m+2,n)}{2\Delta\eta 1},$$

$$\frac{\Delta^2 F^f}{(\Delta\eta^1)^2}(m,n) = \frac{2F(m,n) - 5F(m+1,n) + 4F(m+2,n) - F(m+3,n)}{(\Delta\eta^1)^2},$$

$$\frac{\Delta^2 F^f}{\Delta\eta 1 \Delta\eta 2}(m,n) = -\frac{3\frac{\Delta F^c}{\Delta\eta 2}(m,n) - 4\frac{\Delta F^c}{\Delta\eta 2}(m+1,n) + \frac{\Delta F^c}{\Delta\eta 2}(m+2,n)}{2\Delta\eta 1}$$

and by backward difference operators (denoted by the superscript b) along $m = m_f$:

$$\frac{\Delta F^b}{\Delta\eta 1}(m,n) = \frac{3F(m,n) - 4F(m-1,n) + F(m-2,n)}{2\Delta\eta 1},$$

$$\frac{\Delta^2 F^b}{(\Delta\eta 1)^2}(m,n) = \frac{2F(m,n) - 5F(m-1,n) + 4F(m-2,n) - F(m-3,n)}{(\Delta\eta 1)^2}$$

$$\frac{\Delta^2 F^b}{\Delta\eta^1 \Delta\eta^2}(m,n) = \frac{3\,\frac{\Delta F^c}{\Delta\eta^2}(m,n) - 4\,\frac{\Delta F^c}{\Delta\eta^2}(m-1,n) + \frac{\Delta F^c}{\Delta\eta^2}(m-2,n)}{2\Delta\eta^1} .$$

Notice again that the mixed partials operator involves the successive application of first partial operators (forward or backward with respect to η^1 and central with respect to η^2):

$$\frac{\Delta^2 F^f}{\Delta\eta^1 \Delta\eta^2}(m,n) = \frac{\Delta}{\Delta\eta^1}\left(\frac{\Delta F^c}{\Delta\eta^2}\right)^f , \quad \frac{\Delta^2 F^b}{\Delta\eta^1 \Delta\eta^2} = \frac{\Delta}{\Delta\eta^1}\left(\frac{\Delta F^c}{\Delta\eta^2}\right)^b .$$

Where possible the code takes advantage of this fact to reduce calculations.

Along the boundary $n = n_i$ or n_f the central operators $\frac{\Delta F^c}{\Delta\eta^1}$ and $\frac{\Delta^2 F^c}{(\Delta\eta^1)^2}$ are retained, while the remaining operators are replaced by appropriate forward or backward operators obtained from the above operators by interchanging the roles of m and n, and η^1 and η^2 simultaneously.

At the corners of the domain where $m = m_i$ or m_f and $n = n_i$ or n_f the remaining central operators are replaced by the forward or backward operators appropriate to the common boundaries; in particular, the mixed partials operators involve the successive application of the appropriate forward or backward first partial operators -- for example, at $m = m_i$ and $n = n_f$

$$\frac{\Delta^2 F}{\Delta\eta^1 \Delta\eta^2} = \frac{\Delta}{\Delta\eta_1}\left(\frac{\Delta F^b}{\Delta\eta_2}\right)^f .$$

APPENDIX B
COMMENTS ON ELASTOPLASTIC STRESS EVALUATIONS

In the REPSIL code stresses at time t are calculated in the manner of (finite) incremental plasticity using the incremental strains $\Delta\epsilon^{\alpha}_{\beta}$ which occur between time t - Δt and time t as well as stored values of the stresses at time t - Δt. In the finite difference analysis these calculations are made at every (m,n) mesh node for each k layer. Additionally, where the strain-hardening constitutive option is used, stresses or, more precisely, substresses are calculated for each j sublayer of the mechanical sublayer model (see [2; Section 5.4.2]) and the stresses in each k layer are determined as weighted averages of the stresses in the j sublayers.

At locations where the incremental strains $\Delta\epsilon^{\alpha}_{\beta}$ entail plastic flow, the flow parameter $\Delta\lambda$ of (2.24) must be evaluated as the root of a quadratic equation, see (2.26). An algorithm for dealing with the various types of roots which may occur has been developed by Huffington [3]. If complex roots occur this algorithm subdivides the elastic stress increment, defined by (2.18), into L equal subincrements (for purposes of stress calculation only) in each of which elastic stress increments $\Delta\overset{E}{\sigma}{}^{\alpha}_{\beta}$ /L take place. Stress calculations are performed for each of the L subincrements, consecutively. If at any stage of this process a complex root is obtained, the value of L is increased and the calculation is re-initiated. This procedure is continued until real stresses are determined for time t.

Experience with this procedure has shown that under certain circumstances it may be desirable to use the subdivided increment algorithm even when real roots are obtained for the full increment. An illustration of this is depicted in Figure B.1, where for simplicity it is assumed that σ^1_1 and σ^2_2 are principal stresses (σ^1_2 = 0). The values shown on this figure were taken from an actual computer solution. Starting from the stress state labeled "1", the components of the trial elastic stress vector $\Delta\overset{E}{\sigma}$ were calculated using (2.18) and the correction stress vector - $\Delta\lambda$ $\overset{D}{\sigma}$ was determined by use of (2.23) and (2.28). These calculations which resulted in a real root for $\Delta\lambda$, indicate that the actual stress increment $\Delta\sigma$ for the full time step is represented by the vector joining points "1" and "2". Similarly calculations for the next time step result in a stress increment from point "2" to point "3" and subsequent calculations indicate continued oscillations between the third and fourth quadrants.

Suspecting that these calculations were inaccurate, the calculations for the first time step were performed using the subdivided stress increment algorithm with L=2. At the end of the first half increment, the stress state labeled "1.5" was predicted and, for end of the second

Preceding page blank

half step, the point labeled "2*". Thus, the revised stress increment for the full cycle is presented by the vector $\Delta\sigma^*$. Repeat applications of this algorithm using larger values of the subincrement counter L did not produce any appreciable revision of the stress increment vector $\Delta\sigma^*$ so that this procedure appears to be rapidly convergent.

It is apparent that the inaccuracy associated with the use of L=1 in this example results from the large excursion outside the yield surface performed by the $\Delta\overset{E}{\sigma}$ vector. As a rational approach to arriving at a appropriate value of L to use in a given case, consider that the yield surface is surrounded by concentric ellipsoidal annuli which are labeled L=1,2,3,··· as one progresses radially outward from the yield surface (see Figure B.1). The elastic region within the yield surface is arbitrarily designated L=0. Then the appropriate value of L for the subincrement calculation would be determined by the designation of the ellipsoidal annulus in which the tip of the $\Delta\overset{E}{\sigma}$ vector is located i.e., the trial stress state $\overset{T}{\sigma}$ given by (2.19). The analytical formulation of the criterion for selecting L may be expressed as:

$$L = \mathrm{INT}\left\{\mathrm{YLDFAC}\left(\sqrt{\frac{\overset{T}{\Phi} + \sigma_o^2}{\sigma_o^2}} - 1\right)\right\} + 1$$

where INT signifies the "integer part of" the quantity within the curly brackets. Note that $\overset{T}{\Phi}$, the function defined by (2.20) with the components of σ substituted, is already calculated within the STRESS subroutine to test for yielding. The coefficient YLDFAC is a parameter (not necessarily integral) which permits variation of the "thickness" of the concentric ellipsoidal annuli in stress space. The user should choose YLDFAC to suit his compromise between minimum computer time (YLDFAC=0, for which no subdivision of the elastic stress increment occurs as long as a real $\Delta\lambda$ root is obtained) and maximum accuracy (YLDFAC→ ∞, corresponding to differential subintervals). Although no systematic study of the effect of varying YLDFAC has been conducted, it appears that reasonably accurate stress determinations have been obtained using YLDFAC = 1 with only nominal increases in machine time. For this value of YLDFAC the ellipsoidal annulas interfaces intersect the σ_Y^Y axes at integral multiples of the uniaxial yield stress σ_o.

The REPSIL code now provides an optional printout of L (m,n) arrays for each k layer of the structural shell at specified time cycles. These arrays provide a definition of the regions of the structural shell within which plastic flow is occuring as well as a

qualitative indication of the magnitude of the flow activity. Where a strain-hardening constitutive model is being employed, the largest value of L for any of the sublayers within the kth layer will be printed. Samples of the L array print format are included with the example problems, see Tables 5.8, 5.12, 5.18, 5.20 and 5.22.

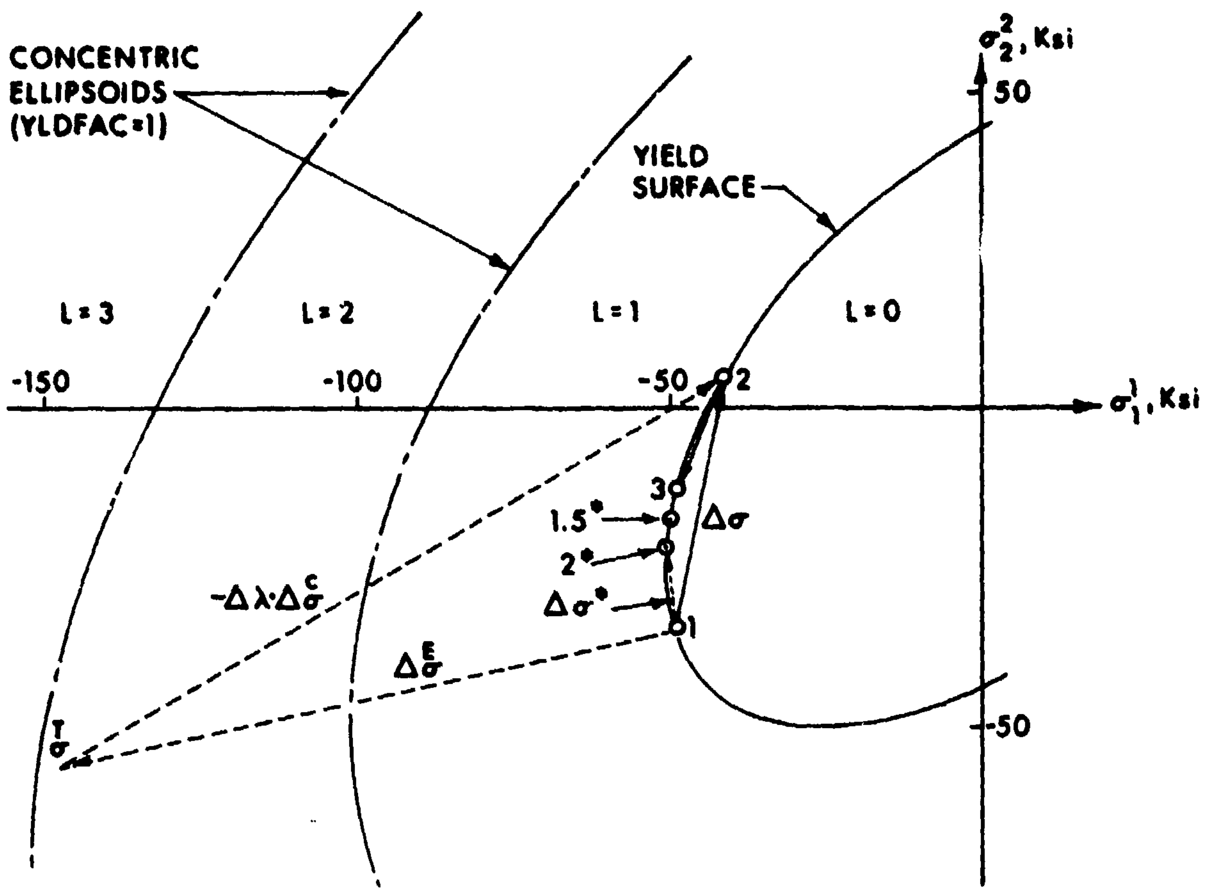


Figure B.1 Graphical Representation of Stress Increments

APPENDIX C.
DEFINITION OF PROGRAM VARIABLES

In this appendix we list the FORTRAN variables used in the program and give a brief description of each, identifying where possible the variable with the symbol used in the body of the report. The variables are grouped according to whether they are integer or real and whother they represent an array or not. Within each group the variables are listed alphabetically. Index notation is applied to certain sets of FORTRAN variables, with Greek indices ranging over the integers 1, 2 and Latin indices over 1, 2, 3, as before. These indices are not subscripted or superscripted. Latin indices are distinguished from letters in the FORTRAN names by not being capitalized. Input variables already described in Section 3.2 are identified by a superscript §.

C.1 Integer Variables

Name	Symbol	Description
I		Index corresponding to surface strain locations, $1 \leq I \leq$ NSTRN.
I1,I2		Dummy indices for elements of the arrays MI1(I) and MI2(I).
IBCE1§ IBCE2§ IBCE3§ IBCE4§		Numbers controlling boundary conditions along edges (input data, card 4).
IFLAG		Control number used in subroutine PDATA.
II		Dimension of DAT(J) array, II=2*NSTRN + 8.
ISR§		Strain rate sensitivity control (input data, card 6).
J		Index corresponding to stress sublayers, $1 \leq J \leq$ NSFL. Also a general dummy index.
J1,J2		Dummy indices for elements of the arrays NI1(I) and NI2(I).
K	k	Index corresponding to layer stations, $1 \leq K \leq$ LAYER.
KD		Dummy argument replacing K.

Preceding page blank

KEY		Control number governing mode of operation of subroutine WRTAPE: 1, write data on tape 2, read data off tape.
KJ		Index corresponding to the Jth stress sublayer of the Kth layer, $1 \le KJ \le KJMAX$.
KJMAX		Total number of stress sublayers in the entire thickness of shell, KJMAX = NSFL*LAYER.
KN		Total number of stress sublayers in the first (K-1) layers, KN = NSFL*(K-1).
L	L	Number of subincrements into which the elastic stress increment $\Delta\sigma^{E\alpha}_{\beta}$ is divided for plastic flow calculations described in Appendix B, $L \le 100$.
LAYER§	K or K_{max}	Total number of layers into which the thickness is divided (input data, card 2).
LC		Counter for the elastic stress subincrement used in plastic flow calculations described in Appendix B, $1 \le LC \le L + 1$.
LINK		Control number used in subroutines STRAIN and PDATA.
LMNK		Counter for the maximum number of subincrements L occuring in a layer, supplying entrees for the array LMAT (M,N,K).
LOAD§		Control number governing mode of loading (input data, card 5).
LPRESS§		Last time step at which pressure is nonconstant (input data, card 5).
M	m	Mesh number in the η^1 direction, $1 \le M \le MM$.
M1		Maximum value of M at which strains and stresses are calculated, a function of the edge conditions along boundary 3: $M1 = \begin{cases} \text{MS, symmetry edge} \\ \text{MM, clamped or hinged edge.} \end{cases}$
MASH§		Number controlling mesh proportion of conical shell (input data, card 14).

MAXC		Final time step (input data, card 3).
MB1		Maximum value of M at which the displacement increments are not modified by clamped edge conditions along boundary 3: $MB1 = \begin{cases} MR, \text{ clamped edge} \\ MS, \text{ symmetry or hinged edge.} \end{cases}$
MD		Dummy argument replacing M.
MDAMP$		Time step at which damping operations begin (input data, card 5).
MESH$		Number of mesh intervals in the η^1 direction (input data, card 2).
MF$		Maximum value of M for points receiving uniform initial velocity VR (input data, card 15).
MI$		Minimum value of M for points receiving uniform initial velocity VR (input data, Card 15).
MM		Total number of mesh points in the η^1 direction.
MQ1, MQ2		Values of M at mesh points bracketing the location at which the displacement U^i is determined, see subroutine STRAIN.
MR		MR = MM-2.
MRITE		Time steps at which restart data is collected by subroutine WRTAPE.
MS		MS = MM-1.
N	n	Mesh number in the η^2 direction, $1 \leq N \leq NN$.
N1		Maximum value of N at which strains and stresses are calculated, a function of the edge conditions along boundary 2: $N1 = \begin{cases} NS, \text{ symmetry edge} \\ NN, \text{ clamped or hinged edge} \end{cases}$
N3D$		Number of time steps at which 3D plots are drawn (input data, card 11).

NB1 Minimum value of N at which the displacement increments are not modified by clamped edge conditions along boundary 4:

$$NB1 = \begin{cases} 2, \text{ hinged edge} \\ 3, \text{ clamped edge.} \end{cases}$$

NB2 Maximum value of N at which the displacement increments are not modified by clamped edge conditions along boundary 2:

$$NB2 = \begin{cases} NR, \text{ clamped edge} \\ NS, \text{ symmetry or hinged edge} \end{cases}$$

NCONT$ Initial time step (input data, card 3).

NCYC NCYC = NCYCLE - 1.

NCYCLE & Time step.

ND Dummy argument replacing N.

NDELP Number of time steps between surface strain prints, replaces NPRINT in program.

NF$ Maximum value of N for points receiving uniform initial velocity VR (input data, card 15).

NI$ Minimum value of N for points receiving uniform initial velocity VR (input data, card 15).

NLP Counter indexing the array JCYNLP(J).

NLPRIN$ Number of times the array LMAT(M,N,K) is printed (input data, card 10).

NMESH$ Number of mesh intervals in the η^2 direction (input data, card 2).

NN Total number of mesh points in the η^2 direction.

NN3D Countor indexing the array NC3DP(J).

NNN Counter indexing the array NCYCH(J).

NPLOT Plotting tape unit number.

NPRINT$ Input on card 8 as the number of time steps between surface strain prints, replaced by NDELP in program. In program NPRINT gives the time step, at which surface strains are printed.

NQ1,NQ2		Values of N at mesh points bracketing the location at which the displacement U^i is determined, see MQ1, MQ2.
NR		NR = NN-2
NRITE$		Number of time steps between collection of restart data (input data, card 3).
NS		NS = NN-1.
NSTLS$	J	Total number of stress sublayers in each layer - a plasticity modelling control (input data, card 6).
NSTRN$		Total number of locations at which surface strain are computed (input data, card 12).
NUMCY$		Number of time steps at which JCHK(J) controlled data and energy-work data are printed (input data, card 9).
NV.		Number of mesh points receiving initial velocities V different from uniform initial velocity VR (input data, card 15).

C. Integer Arrays

JCHK(J)$		Print control (input data, card 8).
JCYNLP(J)$		Time steps at which array LMAT(M,N,K) is printed (input data, card 10).
LMAT(M,NK)	L(m,n)	Matrix of maximum stress subincrements for the Kth layer, see description of L and LMNK, and Appendix B.
MI1(I), MI2(I)		Values of M at mesh points bracketing the Ith surface strain location, MI1(I) $\leq$ PM(I) $\leq$ MI2(I) $\equiv$ MI1(I) + 1.
NC3DP(J)$		Time steps at which 3D plots are obtained (input data, card 11).
NCYCH(J)$		Time steps at which JCHK(J) controlled data and energy-work data are printed (input data, card 9).
NETAG(I)$		Control selecting surface on which Ith surface strain location is situated (input data, card 13).
NI1(I), NI2(I)		Values of N at mesh points bracketing the Ith surface strain location, NI1(I) $\leq$ PN(I) $\leq$ NI2(I) $\equiv$ NI1(I) + 1.

C.3 Real Variables

A11 A12 A22	$A\alpha\beta$	$a_{\alpha\beta}$	Covariant components of middle surface metric.
AA		A	Coefficient of quadratic equation for $\Delta\lambda$.
ALFN			Alphanumeric print indicating surface on which strain are determined.
ANGEL		θ	ANGLB(I) or ANGLE(I) in radians.
AR11 AR12 AR22	$AR\alpha\beta$	$a^{\alpha\beta}$	Contravariant components of middle surface metric.
AVGRAD			Cylinder radius or average cone radius, zero for flat plate.
B		-2B	Coefficient of quadratic equation for $\Delta\lambda$.
B11 B12 B22	$B\alpha\beta$	$b_{\alpha\beta}$	Covariant components of middle surface 2nd fundamental tensor.
BM11 BM12 BM21 BM22	$BM\alpha\beta$	b^{α}_{β}	Mixed components of 2nd fundamental tensor.
BT		b^{γ}_{γ}	Trace of 2nd fundamental tensor.
C		c	Sound speed, $c = \sqrt{\frac{E}{\rho(1-\nu^2)}}$
C1			Program constant used in equations of motion, C1 = C2/(4 + C2).
C1OLD			Old value of C1 in subroutine DESTEP.
C2		$2D\Delta t/\Gamma_o$	Program constant used in damping work calculations.
CA		$\Delta\eta^1\Delta\eta^2$	Program constant used in kinetic energy and work calculations.
CB		$\Delta\eta^1\Delta\eta^2\Delta\zeta/E$	Program constant used in elastic energy calculations.
CINEP		T*	Kinetic energy removed by previous KEA operation.

CINER	T_-	Kinetic energy at time t - 1/2 Δt.
CINES	T_+, T_{++}	Kinetic energy at time t + 1/2 Δt or at time t + 3/2 Δt in subroutines DAMP and DESTEP.
CINES1	T_-	Previous value of CINER in subroutines DAMP and DESTEP.
CINET	T	Kinetic energy at current time t.
CM,CN		Weighing factor accounting for reduced areas at boundarys in summing for the kinetic energy and work.
CS111, CS112, CS122, CS211, CS212, CS222 (CSαβγ)	$\Gamma_{\beta\gamma}^{\alpha}$	Christoffel symbols for the middle surface.
CSM1, CSM2 (CSMα)		Normal components of FNTαi.
D	B^2-AC	Discriminant of quadratic equation for $\Delta\lambda$.
D1, D2, D3 (Di)	U^i	Components of the displacement of the middle surface particle with material coordinate ETAD1, ETAD2.
DA	a	Determinant of the surface metric $a_{\alpha\beta}$.
DA11, DA12, DA22	1/2 $\Delta a_{\alpha\beta}$	One half the incremental change in $a_{\alpha\beta}$ during the time interval from t - Δt to t.
DAMPFS	D	Viscous damping coefficient (input data, card 5).
DB11, DB12, DB22	$\Delta b_{\alpha\beta}$	Incremental change in $b_{\alpha\beta}$ during the time interval from t-Δt to t.
DELB	Δt_B	Critical bending time increment for stability.
DELGAM	$\Delta t^2/\Gamma_o$	Program constant used to calculate TEMP(M,N).
DELIN	Δt	Temporary storage of input DELTAT.
DELM	Δt_M	Critical membrane time increment for stability.

DELMIN		Minimum of DELB and DELM rounded off.
DELR	$\left(\frac{\Delta t^*}{\Delta t}\right)^2$	Factor used in computing Δu^i and T in subroutine DESTEP, see (2.53 and 2.54).
DELS	$\left(\frac{\Delta t^*}{\Delta t}\right)^2 \frac{2\Gamma_o + D\Delta t}{2\Gamma_o + D\Delta t^*}$	Factor used in computing Δu^i and T in subroutine DESTEP, see (2.53 and 2.54).
DELSQ	Δt^2	DELSQ = DELTAT**2.
DELTA	δ	Time constant used in subroutine STRAIN.
DELTAT§	Δt	Time increment (calculated by program or input on card 3).
DEPS11, DEPS12, DEPS21, DEPS22 (DEPSαβ)	$\Delta\varepsilon^\alpha_\beta$	Mixed components of the strain increment $\Delta\varepsilon_{\alpha\beta}$
DETA1, DETA2	$\Delta\eta^\alpha$	Increments in the η^1 and η^2 material coordinates, respectively.
DETAN	$\frac{1}{12}(1 - \frac{1}{K^2})$	Factor used in calculating DELB, see (2.3).
DEFACT§		Factor for terminating damping operations (input data, card 5).
DG	g	Determinant of $g_{\alpha\beta}$.
DN	Δn	Normal component of increment in n^i.
DNL1, DNL2	Δn_α	Tangential components of increment in n^i.
DNR1, DNR2	Δn^α	Contravariant form of Δn_α, above; $\Delta n^\alpha = a^{\alpha\beta}\Delta n_\beta$
DSG11L, DSG12L, DSG21L, DSG22L	$\Delta\sigma^{E\alpha}_{\beta}/L$	Subincrements into which $\Delta\sigma^{E\alpha}_{\beta}$ below is divided.
DSIG11, DSIG12, DSIG21, DSIG22 (DSIGαβ)	$\Delta\sigma^{E\alpha}_{\beta}$	Mixed components of elastic stress increment.
DSQOLD	Δt^2	Old values of DELSQ in subroutine DESTEP.

DUSN, DUSN1, DUSN2	Δu_n	Corrections applied to the normal components of Δu^i along clamped edges and at clamped corners, respectively.
DW	$n^i \Delta u^i$	Normal component of Δu^i.
DX4, DY4	$1/(\Delta\eta^1)^4$, $1/(\Delta\eta^2)^4$	Factors used in calculating DELB,see (2.3)
E§	E	Young's modulus (input data, card 6).
EN	$\Delta W(t-\frac{1}{2}\Delta t)$	Work increment in the time interval t-Δt to t.
ENR	$\frac{\Delta W(t-\Delta t)}{\Delta\eta^1\Delta\eta^2}$	Value of ENS at the time t - Δ.
ENS	$\frac{\Delta W(t)}{\Delta\eta^1\Delta\eta^2}$	Proportional to work increment in the time interval t-1/2 Δt to t+1/2 Δt.
EPSDOT	$\dot{\varepsilon}$	Deviator strain rate used in strain rate dependence law.
EPSR1, EPSR2, GRAMMAR	$\varepsilon 1$, $\varepsilon 2$, 2γ	Intermediate components of strain used in calculating surface strains.
ETA1, ETA2	η^α	Material coordinates η^1 and η^2 of middle surface particles.
ETAD1§, ETAD2§	η^α	Material coordinates of the point at which the displacements U^i are calculated (input data, card 12).
F11, F12, F22	$\frac{M^{\alpha\beta}}{a^{\frac{1}{2}}\Delta\zeta}$	Proportional to bending resultant FMαβ (M,N).
FLAYER	$\left(\frac{K}{2}\right)^2$	Factor used in calculating DELB.
FLOATL	1/L	Fraction of stress subincrement.
FNT11, FNT12, FNT13, FNT21, FNT22, FNT23	$\frac{N^{i\alpha}}{a^{\frac{1}{2}}\Delta\zeta}$	Proportional to stress resultant FNαi (M,N).
FNU§	ν	Poisson's ratio (input data, card 6).
G	$\frac{E}{2(1+\nu)}$	Shear Modulus.

G11, G12, G22	$g_{\alpha\beta}$, $G_{\alpha\beta}$	Used in subroutine STRESS for the metric $g_{\alpha\beta}$ of the lamella ζ distance from the middle surface and in subroutint STRAIN for the initial metric $G_{\alpha\beta}$ on the bounding surface.
GAMMAR		See EPSR1 above.
GAMZ	$\Gamma_0 = \rho h$	Mass per initial middle surface area.
GR	$1/\sqrt{G_{11}G_{22}}$	Time constant computed from $G_{\alpha\beta}$.
GR11, GR12, GR22	$g^{\alpha\beta}$	Inverse of metric $g_{\alpha\beta}$.
GTWO	$\frac{E}{1+\nu}$	Twice the shear modulus G.
H	1/2 h	Half the shell thickness, $H = \frac{THICKN}{2}$.
LENGTH§		Length of plate, cylinder, or cone (input data, card 14).
PHIT	ϕ	Yield function.
PI		The mathematical constant π.
PLAST	W_p	Plastic work.
PM1, PM2, PN1, PN2		PM1 = MI1(I), PM2 = MI2(I), PN1 = NI1(I), PN2 = NI2(I) — Conversion from integer to floating point form.
PRAT	$\frac{E}{1-\nu^2}$	Material constant.
Q11, Q12, Q22	$\frac{Q^{\alpha\beta}}{a^{\frac{1}{2}}\ \Delta\zeta}$	Proportional to membrane components of stress resultant.
QM, QN	m,n	Mesh numbers corresponding to material coordinates ETAD1 and ETAD2.
QM1,QM2, QN1,QN2		Interpolation coefficient used to compute the displacement components Di.
RA	1/a	$RA = \frac{1}{DA}$.
RADI§		Small radius of conical shell (input data, card 14).
RADIUS§		Initial radius of cylindrical shell (input data, card 14).

RADF§		Large radius of conical shell (input data, card 14).
RD11 RD12 RD22	$1/(\Delta\eta^1)^2$ $1/4\Delta\eta^1\Delta\eta^2$ $1/(\Delta\eta^2)^2$	Program constants used in computing second order finite difference derivatives, see Appendix A.
RG	$1/g$	$RG = \frac{1}{DG}$
RHO§	ρ	Initial mass density (input data, card 6).
RN1 RN2 RN3	$\underset{-}{n}^i$	Cartesian components of unit normal SNi (M,N) at the previous time $t - \Delta t$.
RRA	$1/a^{\frac{1}{2}}$	$RRA = \frac{1}{SRA}$.
RSUM		Factor used in rounding off DELMIN.
RTD1 RTD2	$1/2\Delta\eta^1$ $1/2\Delta\eta^2$	Program constants used in computing first order finite difference derivatives, see Appendix A.
SA,SB	α,β	Constants for computing surface strain in θ direction.
SIG11 SIG12 SIG21 SIG22	SIGαβ σ^α_β	Mixed components of stress at time t.
SIG11D SIG12D SIG21D SIG22D	SIGαβD ${}^D\sigma^\alpha_\beta$	Mixed components of plastic flow corrector stress.
SIG11I SIG12I SIG21I SIG22I	SIGαβI $\underset{-}{\sigma}^\alpha_\beta$	Mixed components of stress at time $t - \Delta t$.
SIG11L SIG12L SIG21L SIG22L	SIGαβL ${}^T\sigma^\alpha_\beta$	Mixed components of trial stress.
SIGYSQ	σ_o^2	Square of yield stress or square of yield stress magnified by rate sensitivity factor.
SIGZ§	σ_o	Uniaxial yield stress (input data, card 6).

SNN	$n^i n_-^i$	Scalar product of normals at times t and t - Δt.
SRA	$a^{\frac{1}{2}}$	SRA = SQRT (DA).
SRDEL	$1/\sqrt{1-\delta^2}$	Time constant used in subroutine STRAIN.
SRG	$g^{\frac{1}{2}}$	SRG = SQRT (DG).
SS11, SS12, SS21, SS22 (SS$\alpha\beta$)	σ^α_β	Weighted sum of the mixed components of sublayer stress for given layer, giving the layer stress.
STREN	V	Elastic strain energy.
SUM		Factor used in rounding off DELMIN.
SUMA11, SUMA12, SUMA22	$\sum_k \sigma^{\alpha\beta}$	Sum of components of layer stress SS1MN(k), etc. over all layers.
SUMB11, SUMB12, SUMB22	$\sum_k \sigma^{\alpha\beta}\zeta$	Sum of first moment of components of layer stress SS1MN(k), etc. over all layers.
SUMC11, SUMC12, SUMC22	$\sum_k \sigma^{\alpha\beta}\zeta^2$	Sum of second moments of components of layer stress SS1MN(k), etc. over all layers.
T2	$\frac{1 + 15\nu}{8(\Delta\eta^1\Delta\eta^2)}$	Factor used in calculating DELB, see (2.3).
TA	$\Delta\zeta$	Layer thickness, TA = $\frac{\text{THICKN}}{\text{ZAYER}}$.
TAMBDA	$\Delta\lambda$	Factor measuring amount of plastic flow.
TB	$a^{\frac{1}{2}}\Delta\zeta$	Program constant used in calculating resultants.
TDAMP	W_D	Damping work.
THETA!		Angle subtended by cylindrical and conical panels (input data, card 14).
THICKNS	h	Thickness of shell (input data, card 6).
TIME	t	Current time, TIME = NCYCLE*DELTAT.
TNRG	W	Total external work (due to pressure).
TRD	$1/(2\Delta\eta^1\Delta\eta^2)$	TRD = 2*RD12.

U11 U12 U13 U21 U22 U23	Uαi	u^i_α	First finite difference derivatives of the displacement increment Ui (M,N) with respect to material coordinates ETAα.
U111 U112 U113 U121 U122 U123 U221 U222 U223	Uαβi	$u^i_{\alpha\beta}$	Second finite difference derivatives of the displacement increment Ui (M,N) with respect to material coordinates ETAα.
U1R U2R U3R		Δu^i	Components of displacement increment during the time interval [t - Δt, t].
U1S U2S U3S		Δu^i_+	Components of displacement increment during the time interval [t, t + Δt].
V§		v	Initial velocity at mesh points not receiving uniform initial velocity VR (input data, card 16).
VF1 VF2 VF3		$-P{*}n^i$	Force due to the pressure P (M,N) in the equations of motion.
VM1 VM2 VM3		$\frac{\partial^2(M^{\alpha\beta}n^i)}{\partial\eta\alpha\partial\eta\beta}$	Force due to bending resultants FMαβ(M,N) in the equations of motion.
VN1 VN2 VN3		$\frac{\partial N^{i\alpha}}{\partial\eta^\alpha}$	Force due to stress resultants FNαi(M,N) in the equations of motion.
VR§		v	Uniform initial velocity (input data, card 15).
WIDTH§			Half width of plate (input data, card 12).
Y11 Y12 Y13 Y21 Y22 Y23	Yαi	y^i_α	First finite difference derivatives of the position vector Yi (M,N) with respect to material coordinate ETAα, forming local basis for middle surface.

Y111 Y112 Y113 Y121 Y122 Y123 Y221 Y222 Y223	Yαβi	$y^i_{\alpha\beta}$	Second finite difference derivatives of the position vector Yi (M,N) with respect to material coordinates ETAα.
YLDFAC§			Parameter governing the "thickness" of stress ellipsoidal annuli, see Appendix B (input data, card 2).
YR11 YR12 YR13 YR21 YR22 YR23	YRαi	$a^{\alpha\beta}y^i_\beta$	Dual or reciprocal basis to the basis Yαi.
ZAYER		K	Floating point representation of LAYER.
ZETAK		2ζ	ZETAK = 2 * ZETA(K).

C.4 Real Arrays

A111(I) A121(I) A221(I)	Aαβ1(I)	Initial $a_{\alpha\beta}$ at mesh point (MI1(I),NI1(I)).
A112(I) A122(I) A222(I)	Aαβ2(I)	Initial $a_{\alpha\beta}$ at mesh point (MI1(K), NI2(I)).
A113(I) A123(I) A223(I)	Aαβ3(I)	Initial $a_{\alpha\beta}$ at mesh point (MI2(I),NI1(I)).
A114(I) A124(I) A224(I)	Aαβ4(I)	Initial $a_{\alpha\beta}$ at mesh point (MI2(I),NI2(I)).
ANGLB(I) ANGLE(I)	θ	Angle specifying direction of surface strains E_θ at location I (input data, card 13).
ASA(I),ASB(I)	$2\alpha^2$	Constant for computing surface strain in θ direction.
B111(I) B121(I) B221(I)	Bαβ1(I)	Initial $b_{\alpha\beta}$ at mesh point (MI1(I), NI1(I)).
B112(I) B122(I) B222(I)	Bαβ2(I)	Initial $b_{\alpha\beta}$ at mesh point (MI1(I), NI2(I)).
B113(I) B123(I) B223(I)	Bαβ3(I)	Initial $b_{\alpha\beta}$ at mesh point (MI2(I),NI1(I)).
B114(I) B124(I) B224(I)	Bαβ4(I)	Initial $b_{\alpha\beta}$ at mesh point (MI2(I),NI2(I)).
BSA(I), BSB(I)	$2\beta^2$	Constant for computing surface strain in θ direction.
CSA(I), CSB(I)	$2\alpha\beta$	Constant for computing surface strain in θ direction.
DAT(J)		Array storing energy, displacement and strain plotting data each time step.
DEPS1(K) DEPS2(K) DGAMMA(K)	$\Delta\varepsilon_{11}$ $\Delta\varepsilon_{22}$ $\Delta\varepsilon_{12}$	Elongational components in the η^1 and η^2 directions and shear component of strain increment in the Kth layer.

DM1(I), DM2(I) DN1(I), DN2(I)		Interpolation coefficient used to compute components of surface strain at location I.
DSR(J)§	d_j	Constant in strain rate dependence law (input data, card 7).
EPSANB(I) EPSANG(I)	E_θ	Elongational surface strains in the θ directions at location I.
EPSL1(M,N) EPSL2(M,N) GAMMAL(M,N)	ϵ_{11} ϵ_{22} ϵ_{12}	Elongational components in the η^1 and η^2 directions and shear component of strain on the negative bounding surface at mesh point (M,N).
EPSS1(I) EPSS2(I)	E_1 E_2	Elongational surface strains in the η^1 and η^2 directions at location I.
EPSU1(M,N) EPSU2(M,N) GAMMAU(M,N)	ϵ_{11} ϵ_{22} ϵ_{12}	Elongational components in the η^1 and η^2 directions and shear component of strain on the positive bounding surface at mesh point (M,N).
ETAG1(I)§ ETAG2(I)§	η^α	Material coordinates of the Ith surface strain location (input data, card 13).
FM11(M,N) FM12(M,N) FM22(M,N) FMαβ(M,N)	$M^{\alpha\beta}$	Components of the bending resultant tensor at the mesh point (M,N).
FN11(M,N) FN12(M,N) FN13(M,N) FN21(M,N) FN22(M,N) FN23(M,N) FNαi(M,N)	$N^{i\alpha}$	Components of the stress resultant tensor at the mesh point (M,N).
GAMMAL(M,N)		See EPSL1(M,N) above.
GAMMAU(M,N)		See EPSU1(M,N) above.
GI11(I) GI12(I) GI22(I)	$1/G_{11}$ $2/\sqrt{G_{11}G_{22}}$ $1/G_{22}$	Time constants computed from the initial surface metric $G_{\alpha\beta}$ at strain location I.
P(M,N)	P, P*	The pressure or augmented pressure at mesh point (M,N).
PM(I) PN(I)	m,n	Mesh numbers corresponding to material coordinates ETAG1(I) and ETAG2(I).
PSR(J)§	p_j	Constant in strain rate dependent law (input data, card 7).

SE(J)		Slopes of Jth segments of polygonal approximation to strain hardening stress-strain curve.
SEPS(J)§	ε_j	Strains at corners of polygonal approximation to strain hardening stress-strain curve (input data, card 7).
SIG1(M,N,KJ) SIG2(M,N,KJ) TAU(M,N,KJ)	$\left.\begin{matrix}\sigma^{11}\\ \sigma^{22}\\ \sigma^{12}\end{matrix}\right\}$	Normal and tangential components of stress in the η^1 and η^2 directions at mesh point (M,N), layer K, sublayer J.
SIGZSQ(J)	σ_o^2	Square of yield stress at Jth sublayer.
SN1(M,N) SN2(M,N) SN3(M,N)	n^i	Cartesian components of unit normal to middle surface at mesh point (M,N).
SS1MN(K) SS2MN(K) STMN(K)	$\left.\begin{matrix}\sigma^{11}\\ \sigma^{22}\\ \sigma^{12}\end{matrix}\right\}$	Normal and tangential components of stress in layer K, obtained from the mixed component form SSαβ by raising index.
SSIG(J)§	σ_j	Stresses at corners of polygonal approximation to strain hardening stress-strain curve (input data, card 7).
STMN(K)		See SS1MN(K) above.
TAU(M,N,KJ)		See SIG1(M,N,KJ) above.
TEMP(M,N)	$\Delta t^2/(a_o^{\frac{1}{2}}\Gamma_o)$	Value of time constant at mesh point (M,N).
U1(M,N) U2(M,N) U3(M,N)	Δu^i	Cartesian components of displacement increment undergone by the mesh point (M,N) in the time interval Δt.
WT(J)		Weighing factors used in summing the sublayer stress to obtain layer stress, see SSαβ .
Y1(M,N) Y2(M,N) Y3(M,N)	y^i	Cartesian coordinate of the mesh point (M,N) on the middle surface.
ZETA(K)	ζ	Distance along the normal from middle surface to midpoint of Kth layer.
ZETASQ(K)	ζ^2	Square of ZETA(K).

APPENDIX D
REPSIL PLOTTING PROGRAM

This is an independent program, separate from the REPSIL program. It was written to satisfy the requirement for a visual display of the output from REPSIL and PETROS structural response programs. This plotting program is useful in quickly interpreting results that in tabular form would be extremely difficult if not impossible, to understand.

The program makes use of the Cal Comp Standard Plotting Package SCOOP which is standard software for a large number of computing systems. If SCOOP is not available it can easily be adapted to any particular plotting system.

D.1 Description of Main Program and Subroutines

The main program reads the plotting data tape and controls the flow of information. If the variable IFLAG equals one, data is read and stored in the plotting data arrays. If it equals two, data is read and the program calls subroutine PLOT3D. When IFLAG equals 99999 the program calls subroutine GRAPH.

If the number of cycles are greater than MAXAR-2 an ERROR PRINT will occur indicating the need for enlargement of the following data arrays: TIM(MAXAR), U1(MAXAR), U2(MAXAR), U3(MAXAR), CIN(MAXAR), STC(MAXAR) TNR(MAXAR), DAMPLT(MAXAR), and EPSS1(N), EPSS2(N), where subscript N is equal to NSTRN*MAXAR. If memory is available MAXAR and the plotting data arrays can easily be enlarged.

Subroutine PLOT3D reads one control card and plots a cross-section, profile and isometric view of the middle surface of the shell. The coordinate data points Y1 (M,N), Y2 (M,N) and Y3 (M,N) are mapped into XP, YP for each M and N to form the isometric view of the surface. Every line lying on the surface is plotted, not just those which are visible (cf. Figure 5.8).

The surface is plotted by connecting successive (M,N) mesh points with straight-line segments for each M over all N on the mesh and, in turn, connecting successive (M,N) mesh points for each N over all M.

Input parameters on the control card allow considerable flexibility in plotting the three views. The deflection can be magnified in order to bring out the details of the deformation pattern. The scale to which the plots are drawn can be adjusted in two ways. First, the scaling factor SF is found automatically, all three dimensions are scaled independently in order to fit the surface into a cube. Then the maximum scale of the three dimensions is selected as the desired scaling. If the maximum is less than one, SF is set to one. Second, the user specifies the scaling factor SF on the input control card.

It is recommended that the first way be used with a cube size, SOFC, of 3.5 inches for every new surface to automatically find SF. See Figures 5.3 and 5.8 for examples of the plotted output.

Subroutine GRAPH produces two graphs. First, time vs displacements U1, U2, U3 at the coordinate point (ETAD1, ETAD2), see Figure 5.5 and 5.10. The second graph is energy balance information (time vs kinetic energy, strain energy, total energy and total damping work) which is useful for detection of numerical instabilities and as an indication of when the solution may be terminated, see Figures 5.4 and 5.9.

Subroutine STRGRA produces NSTRN graphs of time vs elongational strain in the η^1 and η^2 directions at the coordinate point (ETAG1, ETAG2) on the inner or outer surface of the shell. The results can be compared with experimental measurements recorded by strain gauge mounted on the surface, see Figures 5.6 and 5.11. See REPSIL input, Section 3.2 for the description of NSTRN.

Subroutine SEDSHL will plot a dashline instead of a solid line.

Subroutine MAXMIN finds the maximum and minimum point for two or more arrays of data on the Y-axis for a given plot.

D.2 Input Plotting Control Card

Variables	Format
DEFLM, SOFC, SF, NPT	3E10.4, I5

DEFLM — Deflection magnification factor.

SOFC — Cube size, automatically finds scale factor SF.

SF — =0, Scaling factor SF is found automatically.
>0, Desired scale factor.
Example: Scale 1/2, punch a real number 2 in columns 21 to 30.

Scale 1/10, punch a real number 10 in columns 21 to 30.

NPT — Mesh point location in the η^2 direction at which a cross-section is desired.

Following are some examples of the plotting control card.

Example 1
DEFLM = 1.0, SOFC = 3.5, SF = 0.0, NPT = 13

Example 2
DEFLM = 3.0, SOFC = 0.0, SF = 2.0, NPT = 33

Example 3

DFFLM = 100.0, SOFC = 0.0, SF = 10.0, NPT = 10

D.3 Description of Variables

Name	Definition
A11, A12, A21 A22, A23	Constants used in the mapping functions XM, YM to rotate and transform [Y1, Y2, Y3] into [XP, YP]
CIN(1500)	The array of numbers, corresponding to kinetic energy used in the energy balance plot.
DAMPLT(1500)	The array of numbers, corresponding to total damping work used in the energy balance plot.
DAT(20)	Array used in the plotting data tape input list.
EPSS1(9000)	The array of numbers, corresponding to the strains in η^1 direction.
EPSS2(9000)	The array of numbers, corresponding to the strains in the η^2 direction.
ETAD1, ETAD2	See Section 3.2 and Appendix C.
ETAG1(6), ETAG2(6)	See Section 3.2 and Appendix C.
HEAD1(4)	First line of the title that appears on the first isometric plot.
HEAD2(3)	Second line of the title that appears on the first isometric plot.
HEAD3(2)	Title that appears on every isometric plot.
I1	Number of mesh points in the η^1 direction.
I2	Number of mesh points in the η^2 direction.
IBCE3	Boundary condition value.
IBUF(1000)	Array used by the Cal Comp basic software package
IPEN	Control for pen during movement.

IFLAG	A flag. When it equals 1, the program will read data for the time wise plots. When it equals 2, the program will read data for the cross-section, profile and isometric views. When it equals 99999, the program will plot the time wise plots.
M1	Same as I1.
MAXAR	Size of plotting data arrays.
MAXA	MAXA = MAXAR-2
NCYCL	The number of time steps for each curve on the time wise plots.
NCYCLE	Time cycle counter.
NETAG(6)	Zero or one, corresponding to inner or outer surface used in the title of the strain plots.
NPLOT	Magnetic tape input unit number.
NSTRN	Number of strain plots.
PA(4)	Array that stores the dash line pattern.
PM(6), PN(6)	The arrays which stores the mesh location (M,N) that appear in the title on the strain plots.
QM, QN	Mesh location (M,N) that appear in the title on the vector displacement plot.
SF	Scale factor.
SOFC	Size of cube.
STC(1500)	The array of numbers corresponding to strain energy used in the energy balance plot.
TE_1, TE_2, TE_3 XYZ	Values used in the mapping functions.
TI	$TI = TIME \times 1.0 \times 10^6$
TIM (1500)	The array of numbers corresponding to time used in the time wise plots.
TIME	Current values of time.
TNR (1500)	The array of numbers corresponding to total energy used in the energy plot.

U_1 (1500), U_2 (1500), U_3 (1500)	The arrays of numbers corresponding to the vector displacement U1, U2, U3.
X(25, 35), Y(25, 35), Z(25, 35)	The arrays of numbers corresponding to the initial shape, Y1(M,N), Y2(M,N), Y3(M,N).
X1(90)	The array of XP values for plotting the cross-section and profile views.
X2(90)	The array of YP values for plotting the cross-section and profile views.
XBAR, YBAR	Board coordinate point on the plotting paper in inches, measured from the lower left corner of the page.
XL	Length of X-axis in inches.
XM (A1, A2)	X-axis mapping function used in isometric view.
XMAX	Maximum X-coordinate.
XMIN	Minimum X-coordinate.
XP,YP	X,Y coordinates on the plotting surface.
XPAGE	Plotting page length.
Y1(25,35),Y2(25,35), Y3(25,35)	See Appendix C.
YM(A1,A2,A3)	Y-axis mapping function used in isometric view.
YMAX, YMIN	Maximum and Minimum Y-coordinate.
YP	See XP.
ZMAX, ZMIN	Maximum and Minimum Z-coordinate.

D.4 FORTRAN Listing of Plotting Program

A complete FORTRAN listing is given in the following order.

1. MAIN Program
2. PLOT 3D
3. GRAPH
4. STRGRA
5. MAXMIN
6. SCDSHL

```
C      REPSIL PLOTTING PROGRAM   (CALCOMP SOFTWARE PACKAGE SCOUP)
      COMMON ETAG1(6),ETAG2(6),PM(6),PN(6),NETAG(6),NSTRN,MAXAR
      COMMON Y1(25,35),Y2(25,35),Y3(25,35),X(25,35),Y(25,35),Z(25,35)
      COMMON DAT(20),NCYCLE,TIME,IBCE3,ETAD1,ETAD2,QM,QN,NCYCL
      COMMON TIM(1500),U1(1500),U2(1500),U3(1500),CIN(1500),STC(1500),
     1TNR(1500),DAMPLT(1500),EPSS1(9000),EPSS2(9000)
C
      NCYCL=0
      NPLOT=3
      REWIND NPLOT
      MAXAR=1500
      MAXA=MAXAR-2
C
      READ(NPLOT) IBCE3,ETAD1,ETAD2,QM,QN,NSTRN
      READ(NPLOT) (ETAG1(I),ETAG2(I),PM(I),PN(I),NETAG(I),I=1,NSTRN)
      READ(NPLOT) NCYCLE,TIME,M1,N1,((Y1(M,N),Y2(M,N),Y3(M,N),M=1,M1),
     1N=1,N1)
      II=2*NSTRN+8
C      SAVE INITIAL SHAPE NEEDED FOR DEFLECTION MAGNIFICATION IN PLOT3D
      DO 5 M=1,M1
      DO 5 N=1,N1
      X(M,N)=Y1(M,N)
      Y(M,N)=Y2(M,N)
      Z(M,N)=Y3(M,N)
    5 CONTINUE
      CALL PLOT3D(M1,N1)
C
   10 READ(NPLOT) IFLAG
      IF(IFLAG .EQ. 99999)GOTO 30
      GOTO(20,25),IFLAG
C
   20 READ (NPLOT) NCYCLE,(DAT(I),I=1,II)
      NCYCL=NCYCL+1
      TIM(NCYCL)=DAT(1)
      U1(NCYCL)=DAT(2)
      U2(NCYCL)=DAT(3)
      U3(NCYCL)=DAT(4)
      CIN (NCYCL)=DAT(5)
      STC (NCYCL)=DAT(6)
      TNR (NCYCL)=DAT(7)
      DAMPLT(NCYCL)=DAT(8)
      DO 22 I=9,II,2
      J=NCYCL+MAXAR*(I-9)/2
      EPSS1(J)=DAT(I)
      EPSS2(J)=DAT(I+1)
   22 CONTINUE
      IF(NCYCL .GE. MAXA)GOTO 28
      GOTO 10
C
   25 READ(NPLOT) NCYCLE,TIME,M1,N1,((Y1(M,N),Y2(M,N),Y3(M,N),M=1,M1),
     1N=1,N1)
      CALL PLOT3D(M1,N1)
      GOTO 10
C
   28 WRITE(6,100) NCYCL
   30 CALL GRAPH
      CALL EXIT
C
  100 FORMAT(15H1ERROR NCYCL = ,I5,29H ENLARGE PLOTTING DATA ARRAYS)
      END
```

```
      SUBROUTINE PLOT3D (I1,I2)
      DIMENSION IBUF(1000),X1(90),X2(90),HEAD1(4),HEAD2(3),HEAD3(2)
      COMMON(USE MAIN)
      COMMON    /PLOTO/  YNP,XDIST,PA(4)
      DATA I/0/,A11/.707107/,A12/.707107/,A21/-.408248/,A22/.408248/,
     1A23/.816497/
      DATA (HEAD1(IK),IK=1,4)/10HCROSS SECT,10HION DEFLEC,10HTION AT N ,
     12H. /
      DATA (HEAD2(IK),IK=1,3)/10HDEFLECTION,10H MAGNIFIED,1H /
      DATA (HEAD3(IK),IK=1,2)/10H MICROSECO,3HNDS/
      DATA (PA(J),J=1,4)/.1,.1,.1,.1/
C
      XM(A1,A2)=SF*(A11*A1+A12*A2)
      YM(A1,A2,A3)=SF*(A21*A1+A22*A2+A23*A3)
C
      IF(I.EQ.0)GOTO 10
      I=I+1
      XBARN=-4.25
      YBAR=10.0
      YNP=YNP+10.0
      IF(I.LE.3)GOTO 30
      CALL PLOT (XDIST,-YNP,-3)
      GOTO 20
   10 XPAGE=200.0
      READ (5,11) DEFLM,SOFC,SF,NI2
      XDIST=6.75
   11 FORMAT(3E10.4,I5)
C     --------------------- SCALE FACTOR FOR 3D PLOT -----------------
      XMAX=X(2,1)
      XMIN=X(2,1)
      YMAX=Y(2,1)
      YMIN=Y(2,1)
      ZMAX=Z(2,1)
      ZMIN=Z(2,1)
      DO 12 M=2,I1
      DO 12 N=1,I2
      XMAX=AMAX1(X(M,N),XMAX)
      XMIN=AMIN1(X(M,N),XMIN)
      YMAX=AMAX1(Y(M,N),YMAX)
      YMIN=AMIN1(Y(M,N),YMIN)
      ZMAX=AMAX1(Z(M,N),ZMAX)
      ZMIN=AMIN1(Z(M,N),ZMIN)
   12 CONTINUE
      IF(SF .NE. 0.0)GOTO 13
      XS=(XMAX-XMIN)/SOFC
      YS=(YMAX-YMIN)/SOFC
      ZS=(ZMAX-ZMIN)/SOFC
      SF=AMAX1(XS,YS,ZS)
      SF=AINT(SF)
      IF(SF .LT. 1.0) SF=1.0
   13 CALL PLOTS (IBUF,1000,XPAGE)
      CALL SYMBOL (1.0,1.0,.1,HEAD1,0.0,32)
      CALL NUMBER (999.,999.,.1,NI2,0.0,2HI5)
      CALL SYMBOL (1.0,0.8,.1,HEAD2,0.0,21)
      CALL NUMBER (999.,999.,.1,DEFLM,0.0,1)
      CALL SYMBOL (1.0,0.6,.1,8HSCALE 1/,0.0,8)
      CALL NUMBER (999.,999.,.1,SF,0.0,-1)
      SF=1.0/SF
C     ------------------------------------------------------------
   20 YBAR=4.3
```

```
      XBARN=1.0
      YNP=YBAR
      I=1
   30 XBAR=XBARN
C     ------ -------------- MIDSECTION DEFLECTION --------------------
      CALL PLOT (XBAR,YBAR,-3)
      CALL PLOT (0.0,-1.0,3)
      CALL PLOT (0.0,1.0,2)
      CALL PLOT (0.0,0.0,3)
      CALL PLOT (1.0,0.0,2)
      K=0
      DO 90 M=2,I1
      K=K+1
      X1(K)=SF*X(M,NI2)
      X2(K)=SF*Z(M,NI2)
   90 CONTINUE
      X1(K+1)=0.0
      X1(K+2)=1.0
      X2(K+1)=0.0
      X2(K+2)=1.0
      CALL SCDSM (X1,X2,K,1,PA,4)
      CALL LINE (X1,X2,K,1,-1,2)
      K=0
      DO 95 M=2,I1
      K=K+1
      X1(K)=SF*(X(M,NI2)+DEFLM*(Y1(M,NI2)-X(M,NI2)))
      X2(K)=SF*(Z(M,NI2)+DEFLM*(Y3(M,NI2)-Z(M,NI2)))
   95 CONTINUE
      CALL LINE (X1,X2,K,1,1,4)
C     ------------------------------------------------------------------
      XB=2.25
      YB=-2.0
      CALL SYMBOL (XB,YB,.1,6HCYCLE ,0.0,6)
      CALL NUMBER (999.,999.,.1,NCYCLE,0.0,2HI5)
      YB=YB-0.2
      TI=TIME*1.00E 06
      CALL NUMBER (XB,YB,.1,TI,0.0,3)
      CALL SYMBOL (999.,999.,.1,HEAD3,0.0,13)
C     ----------------- CENTER LINE DEFLECTION PROFILE ----------------
      CALL PLOT (0.0,3.0,-3)
      XB=SF*Y(2,I2)
      YB=0.0
      CALL PLOT (XB,YB,2)
      XB=0.0
      YB=-SF*Z(2,1)
      CALL PLOT (XB,YB,3)
      YB=SF*Z(2,1)
      CALL PLOT (XB,YB,2)
C
      L=2
      DO 85 J=1,2
      K=0
      DO 82 N=1,I2
      K=K+1
      X1(K)=SF*Y(L,N)
      X2(K)=SF*Z(L,N)
   82 CONTINUE
      X1(K+1)=0.0
      X1(K+2)=1.0
      X2(K+1)=0.0
```

```
      X2(K+2)=1.0
      CALL SCDSML (X1,X2,K,1,PA,4)
      CALL LINE (X1,X2,K,1,-1,2)
      K=0
      DO 83 N=1,I2
      K=K+1
      X1(K)=SF*(Y(L,N)+DEFLM*(Y2(L,N)-Y(L,N)))
      X2(K)=SF*(Z(L,N)+DEFLM*(Y3(L,N)-Z(L,N)))
   83 CONTINUE
      CALL LINE (X1,X2,K,1,1,4)
      IF(IBCE3 .NE. 2)GOTO 86
      L=I1
   85 CONTINUE
C     ---------------------------- DRAW AXIS --------------------------
   86 CALL PLOT (4.25,-3.0,-3)
      CALL PLOT (0.0,-1.0,3)
      CALL PLOT (0.0,1.0,2)
      CALL PLOT (0.0,0.0,3)
      CALL PLOT (-A12,-A22,2)
      CALL PLOT (0.0,0.0,3)
      CALL PLOT (A11,A21,2)
C     ---------------------------------------------------------------
      DO 50 M=2,I1
      IPEN=3
      DO 40 N=1,I2
      TE1=X(M,N)+DEFLM*(Y1(M,N)-X(M,N))
      TE2=Y(M,N)+DEFLM*(Y2(M,N)-Y(M,N))
      TE3=Z(M,N)+DEFLM*(Y3(M,N)-Z(M,N))
      XP=XM(TE1,TE2)
      YP=YM(TE1,TE2,TE3)
   35 CALL PLOT (XP,YP,IPEN)
      IF(IPEN .LT. 3)GOTO 40
      IPEN=2
      GOTO 35
   40 CONTINUE
   50 CONTINUE
      DO 70 N=1,I2
      IPEN=3
      DO 60 M=2,I1
      TE1=X(M,N)+DEFLM*(Y1(M,N)-X(M,N))
      TE2=Y(M,N)+DEFLM*(Y2(M,N)-Y(M,N))
      TE3=Z(M,N)+DEFLM*(Y3(M,N)-Z(M,N))
      XP=XM(TE1,TE2)
      YP=YM(TE1,TE2,TE3)
   45 CALL PLOT (XP,YP,IPEN)
      IF(IPEN .LT. 3)GOTO 60
      IPEN=2
      GOTO 45
   60 CONTINUE
   70 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE GRAPH
      DIMENSION SYM1(4),SYM2(3),ETA1(3),ETA2(3),X1(4),X2(4)
      COMMON (USE MAIN)
      COMMON    /PLOTD/  YNP,XDIST,PA(4)
      DATA(SYM1(I),I=1,4)/10HCOMPONENT ,10HOF VECTOR ,10HDISPLACEME,2HNT
     1/
      DATA(ETA1(I),I=1,3)/10HETA1 =    ,10H      (   ,4H   )/,
     1    (ETA2(I),I=1,3)/10HETA2 =    ,10H      (   ,4H   )/,
      DATA(SYM2(I),I=1,3)/10HENERGY BAL,10HANCE, POUN,8HD-INCHES/
C
      XDIST=8.0
      CALL PLOT (XDIST,-YNP,-3)
      XBAR=0.0
      YNP=2.0
      XL=8.0
      YL=6.0
      N=NCYCL
      CALL SCALE (TIM,XL,N,1)
      XMIN=TIM(N+1)
      XS=TIM(N+2)
C     ------------- GRAPH ONE  (VECTOR DISPLACEMENT) ----------------
      CALL MAXMIN (U1,YMIN,YMAX,1,N,1)
      CALL MAXMIN (U2,YMIN,YMAX,1,N,2)
      CALL MAXMIN (U3,YMIN,YMAX,1,N,2)
      X1(1)=YMIN
      X1(2)=YMAX
      CALL SCALE (X1,YL,2,1)
      YMIN=X1(3)
      YS=X1(4)
      U1(N+1)=YMIN
      U1(N+2)=YS
      U2(N+1)=YMIN
      U2(N+2)=YS
      U3(N+1)=YMIN
      U3(N+2)=YS
      CALL PLOT (XBAR,YNP,-3)
      CALL AXIS (0.0,0.0,4HTIME,-4,XL,0.0,XMIN,XS)
      CALL AXIS (0.0,0.0,SYM1,32,YL,90.0,YMIN,YS)
      CALL LINE (TIM,U1,N,1,0,0)
      TX=TIM(N)/XS+.2
      TY=(U1(N)-YMIN)/YS
      CALL SYMBOL (TX,TY,.1,2HU1,0.0,2)
      CALL LINE (TIM,U2,N,1,0,0)
      TY=(U2(N)-YMIN)/YS
      CALL SYMBOL (TX,TY,.1,2HU2,0.0,2)
      CALL LINE (TIM,U3,N,1,0,0)
      TY=(U3(N)-YMIN)/YS
      CALL SYMBOL (TX,TY,.1,2HU3,0.0,2)
      CALL SYMBOL (2.1,-1.0,.1,8HLOCATION,0.0,8)
      CALL SYMBOL (3.0,-1.15,.4,1H ,0.0,1)
      CALL SYMBOL (5.3,-1.15,.4,1H ,0.0,1)
      CALL SYMBOL (3.1,-0.9,.1,ETA1,0.0,24)
      CALL NUMBER (3.8,999.,.1,ETAD1,0.0,3)
      CALL NUMBER (4.8,999.,.1,QM,0.0,3)
      CALL SYMBOL (3.1,-1.1,.1,ETA2,0.0,24)
      CALL NUMBER (3.8,999.,.1,ETAD2,0.0,3)
      CALL NUMBER (4.8,999.,.1,QN,0.0,3)
C     ---------------- GRAPH TWO  (ENERGY BALANCE) -----------------
      XBAR=0.0
      YNP=10.0
```

```
      CALL MAXMIN (CIN,YMIN,YMAX,1,N,1)
      CALL MAXMIN (STC,YMIN,YMAX,1,N,2)
      CALL MAXMIN (TNR,YMIN,YMAX,1,N,2)
      CALL MAXMIN (DAMPLT,YMIN,YMAX,1,N,2)
      X1(1)=YMIN
      X1(2)=YMAX
      CALL SCALE (X1,YL,2,1)
      YMIN=X1(3)
      YS=X1(4)
      CIN(N+1)=YMIN
      CIN(N+2)=YS
      STC(N+1)=YMIN
      STC(N+2)=YS
      TNR(N+1)=YMIN
      TNR(N+2)=YS
      DAMPLT(N+1)=YMIN
      DAMPLT(N+2)=YS
      CALL PLOT (XBAR,YNP,-3)
      CALL AXIS (0.0,0.0,4HTIME,-4,XL,0.0,XMIN,XS)
      CALL AXIS (0.0,0.0,SYM2,28,YL,90.0,YMIN,YS)
      CALL LINE (TIM,CIN,N,1,0,0)
      CALL LINE (TIM,STC,N,1,0,0)
      CALL LINE (TIM,TNR,N,1,0,0)
      CALL LINE (TIM,DAMPLT,N,1,0,0)
C     -------------------- SURFACE STRAIN GRAPHS ---------------------
      DO 100 I=1,NSTRN
      J=1+MAXAR*(I-1)
      CALL STRGRA (TIM,EPSS1,EPSS2,ETAG1(I),ETAG2(I),PM(I),PN(I),
     1NETAG(I),J,N)
  100 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE STRGRA (X,Y,Z,ETA1,ETA2,PM1,PN1,NETA,J,N)
C     X IS THE NAME OF THE ARRAY CONTAINING THE X COORDINATE DATA.
C     Y,Z IS THE NAMES OF THE ARRAYS CONTAINING THE Y COORDINATE DATA.
C     ETA1,ETA2 LOCATION POINT USED IN THE TITLE.
C     PM1,PN1 MESH LOCATION USED IN THE TITLE.
C     NETA IS THE CONTROL FOR INNER OR OUTER USED IN THE TITLE.
C     J IS THE LOCATION OF THE FIRST DATA POINT IN ARRAYS Y AND Z FOR
C     EACH STRAIN PLOT.
C     N IS THE NUMBER OF DATA POINTS.
      DIMENSION X(1),Y(1),Z(1),SYM1(3),SYM2(3),SYM3(2),SYM4(2),X1(4),
     1X2(4)
      COMMON    /PLOTD/  YNP,XDIST,PA(4)
      DATA(SYM1(K),K=1,3)/10HETA1 =     ,10H      (    ,4H    )/,
     1    (SYM2(K),K=1,3)/10HETA2 =     ,10H      (    ,4H    )/,
     2    (SYM3(K),K=1,2)/10HETA2 COMPO,4HNENT/,
     3    (SYM4(K),K=1,2)/10HETA1 COMPO,4HNENT/
C
      IF(I.EQ.0)GOTO 10
      I=I+1
      YNP=YNP+10.0
      YBAR=10.0
      IF(I.LE.3)GOTO 25
      YNP=YNP-10.0
      CALL PLOT (XDIST,-YNP,-3)
      GOTO 20
   10 XDIST=13.0
      CALL PLOT (XDIST,-YNP,-3)
      XL=8.0
      YL=6.0
      CALL SCALE (X,XL,N,1)
      XMIN=X(N+1)
      XS=X(N+2)
   20 XBAR=0.0
      YNP=0.0
      YBAR=0.0
      I=1
   25 CALL MAXMIN (Y,YMIN,YMAX,J,N,1)
      CALL MAXMIN (Z,YMIN,YMAX,J,N,2)
      X1(1)=YMIN
      X1(2)=YMAX
      CALL SCALE (X1,YL,2,1)
      YMIN=X1(3)
      YS=X1(4)
      JM=J+N
      JS=J+N+1
      Y(JM)=YMIN
      Y(JS)=YS
      Z(JM)=YMIN
      Z(JS)=YS
      CALL PLOT (XBAR,YBAR,-3)
      CALL AXIS (0.0,0.0,4HTIME,-4,XL,0.0,XMIN,XS)
      CALL AXIS (0.0,0.0,10HSTRAIN ( ),10,YL,90.0,YMIN,YS)
      CALL SYMBOL (1.8,-1.0,.1,8HLOCATION,0.0,8)
      CALL SYMBOL (2.7,-1.15,.4,1H ,0.0,1)
      CALL SYMBOL (5.0,-1.15,.4,1H ,0.0,1)
      IF(NETA .NE. 0)GOTO 30
      CALL SYMBOL (5.3,-1.0,.1,5HOUTER,0.0,5)
      GOTO 35
   30 CALL SYMBOL (5.3,-1.0,.1,5HINNER,0.0,5)
   35 CALL SYMBOL (2.8,-0.9,.1,SYM1,0.0,24)
```

```
      CALL NUMBER (3.5,999.,.1,ETA1,0.0,3)                              STRG  61
      CALL NUMBER (4.5,999.,.1,PM1,0.0,3)                               STRG  62
      CALL SYMBOL (2.8,-1.1,.1,SYM2,0.0,24)                             STRG  63
      CALL NUMBER (3.5,999.,.1,ETA2,0.0,3)                              STRG  64
      CALL NUMBER (4.5,999.,.1,PN1,0.0,3)                               STRG  65
      X1(1)=2.9                                                         STRG  66
      X2(1)=-1.3                                                        STRG  67
      X1(2)=3.4                                                         STRG  68
      X2(2)=-1.3                                                        STRG  69
      X1(3)=0.0                                                         STRG  70
      X2(3)=0.0                                                         STRG  71
      X1(4)=1.0                                                         STRG  72
      X2(4)=1.0                                                         STRG  73
      CALL LINE (X1,X2,2,1,0,0)                                         STRG  74
      CALL SYMBOL (3.6,-1.3,.1,SYM4,0.0,14)                             STRG  75
      X2(1)=-1.5                                                        STRG  76
      X2(2)=-1.5                                                        STRG  77
      CALL SCDSHL (X1,X2,2,1,PA,4)                                      STRG  78
      CALL SYMBOL (3.6,-1.5,.1,SYM3,0.0,14)                             STRG  79
      CALL LINE (X,Y(J),N,1,0,0)                                        STRG  80
      CALL SCDSHL (X,Z(J),N,1,PA,4)                                     STRG  81
      RETURN                                                            STRG  82
      END                                                               STRG  83
```

```
      SUBROUTINE MAXMIN (A,AMIN,AMAX,J,N,KEY)
      DIMENSION A(1)
      GOTO (10,20),KEY
   10 J1=J
      J2=J+N-1
      AMIN=A(J1)
      AMAX=A(J2)
   20 DO 100 I=J1,J2
      AMAX=AMAX1(AMAX,A(I))
      AMIN=AMIN1(AMIN,A(I))
  100 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE SCDSHL(X,Y,M,INC,PA,N)
C(I)  X,Y ARE THE NAMES OF THE ARRAYS CONTAINING THE X AND Y
C       COORDINATES, AND THE SCALING PARAMETERS.  (SEE 'SCALE'.)
C(I)  M IS THE NUMBER OF POINTS IN THE X AND Y ARRAYS.  THIS DOES
C       NOT INCLUDE THE TWO EXTRA LOCATIONS FOR THE SCALING PARAMETERS.
C(I)  INC IS THE INCREMENT THAT THE SCDSHL SUBROUTINE IS TO USE
C       IN GETTING DATA FROM THE X AND Y ARRAYS.  (SEE 'SCALE'.)
C(I)  PA IS THE NAME OF THE LINEAR ARRAY WHICH CONTAINS THE ELEMENTS
C       OF THE PATTERN WHICH IS TO BE REPEATED UNTIL THE CURVE
C       IS DRAWN.
C         PA(I), FOR I=1,3,5,......, CORRESPOND TO THE DASHES IN
C           THE PATTERN, WHILE
C         PA(I), FOR I=2,4,6,......, CORRESPOND TO THE SPACES
C           BETWEEN THE DASHES.
C(I)  N IS THE NUMBER OF ELEMENTS IN THE PATTERN DESCRIPTION.
      DIMENSION PA(1),X(1),Y(1)
      NP=M*INC+1
      NQ=NP+INC
      DX=1.0/X(NQ)
      FX=X(NP)
      DY=1.0/Y(NQ)
      FY=Y(NP)
      K=0
      DLN=0.0
      DX2=(X(1)-FX)*DX
      DY2=(Y(1)-FY)*DY
      I=1
      J=1
      DTD=PA(1)
      CALL PLOT(DX2,DY2,3)
      GOTO 30
   20 DSI=DY2-DY1
      DCO=DX2-DX1
      DLN=SQRT(DSI*DSI+DCO*DCO)
      K=MOD(J,2)
      IF(DLN.LE.DTD)GOTO 30
      DX1=DX1+DTD*DCO/DLN
      DY1=DY1+DTD*DSI/DLN
      CALL PLOT(DX1,DY1,3-K)
      J=J+1
      IF(J.GT.N)J=1
      DTD=PA(J)
      GOTO 20
   30 IF(K.NE.0)CALL PLOT(DX2,DY2,2)
      DX1=DX2
      DY1=DY2
      I=I+INC
      IF(I.GT.M) RETURN
      DX2=(X(I)-FX)*DX
      DY2=(Y(I)-FY)*DY
      DTD=DTD-DLN
      GO TO 20
      END
```

APPENDIX E.
FORTRAN LISTING OF THE REPSIL PROGRAM

The listing is given in the following order, consisting of Main program and 22 subroutines.

1. MAIN Program	12. BOUNDU
2. START	13. ABINIT
3. INVEL	14. SYMTRY
4. POSITN	15. KINET
5. DGEOM	16. PWORK
6. GRAD	17. DAMP
7. STRESS	18. DESTEP
8. RESULT	19. PDATA
9. MOTION	20. PRESS*
10. WRTAPE	21a. INGEOM (Flat Plate)
11. STRAIN	21b. INGEOM (Full Cylinder)
	21c. INGEOM (Cone)

There are three nonstandard FORTRAN statements used:

1. COMMON (USE MAIN) is used instead of repeating long COMMON statements in the subroutines that appear in the MAIN program.

2. CALL SKIPFILE (t,n) is used in subroutine PDATA to move tape t forward n file marks. The next READ or WRITE statement will begin with the information after the file mark.

3. CALL BACKFILE (t,n) is used in subroutine PDATA to move tape t backward n file marks. The next READ statement will begin with the information after the file mark, the WRITE will erase the file mark.

---

*Subroutine PRESS gives the pressure loading applied in the flat plate example, Section 5.1.

Preceding page blank

```
C       MAIN PROGRAM
C
C               TAPE 1 RESTART INFORMATION
C               TAPE (NPLOT) PLOTTING DATA
      COMMON Y1(23,34),Y2(23,34),Y3(23,34),U1(23,34),U2(23,34),U3(23,34)
     1,FM11(23,34),FM12(23,34),FM22(23,34),FN11(23,34),FN12(23,34),
     2FN13(23,34),FN21(23,34),FN22(23,34),FN23(23,34),SN1(23,34),
     3SN2(23,34),SN3(23,34),TEMP(23,34),P(23,34),EPSL1(23,34),
     4EPSL2(23,34),GAMMAL(23,34),EPSU1(23,34),EPSU2(23,34),GAMMAU(23,34)
     5,SIG1(23,34,12),SIG2(23,34,12),TAU(23,34,12),LMAT(23,34, 4),
     6DEPS1(4),DEPS2(4),DGAMMA(4),ZETA(4),SS1MN(4),SS2MN(4),STMN(4),
     7ZETASQ(4),NCYCH(50),NC3DP(50),JCYNLP(50)
      COMMON MM,MR,MS,NN,NR,NS,N1,M1,RD11,RD12,RD22,RTD1,RTD2,MB1,
     1DETA1,DETA2,TRD,NN3D,DAMPF,DFACT,MDAMP,TDAMP,LOAD,DELGAM
      COMMON E,FNU,G,PRAT,SIGZ,GAMZ,H,LAYER,DELTAT,TIME,LPRESS,GTWO,
     1NNN,NCYCLE,NRITE,NCONT,NSTRN,CINER,CINES,CINEP,C1,C2,NPLOT,
     2DELSQ,TA,MAXC,MRITE,CA,CB,CINET,STREN,PLAST,TNRG,NB1,NB2,ENS,ENR
      COMMON NPRINT,NDELP,MESH,NMESH,IBCE1,IBCE2,IBCE3,IBCE4,ISR,NSFL,
     1KJMAX,YLDFAC,NLP
      COMMON MI1(6),MI2(6),NI1(6),NI2(6),DM1(6),DM2(6),DN1(6),DN2(6),
     1PM(6),PN(6),ETAG1(6),ETAG2(6),ANGLE(6),ANGLB(6),NETAG(6),EPSS1(6),
     2EPSS2(6),JCHK(3)
      COMMON QM,QN,MQ1,MQ2,NQ1,NQ2,QM1,QM2,QN1,QN2,D1,D2,D3,ETAD1,ETAD2
      COMMON GI11(6),GI22(6),GI12(6),ASA(6),BSA(6),CSA(6),ASB(6),BSB(6),
     1CSB(6),B111(6),B121(6),B221(6),A111(6),A121(6),A221(6),B112(6),
     2B122(6),B222(6),A112(6),A122(6),A222(6),B113(6),B123(6),B223(6),
     3A113(6),A123(6),A223(6),B114(..,B124(6),B224(6),A114(6),A124    ),
     4A224(6),DSR(3),PSR(3),SSIG(3),SEPS(3),SE(3),SIGZSQ(3),WT(3)
      COMMON Y11,Y12,Y13,Y21,Y22,Y23,U11,U12,U13,U21,U22,U23,
     1        Y111,Y112,Y113,Y121,Y122,Y123,Y221,Y222,Y223,
     2        U111,U112,U113,U121,U122,U123,U221,U222,U223
      COMMON A11,A12,A22,SRA,CS111,CS112,CS122,CS211,CS212,CS222,
     1        B11,B12,B22,BT,BM11,BM12,BM21,BM22,CINES1
C
      NPLOT=3
      NLP=1
      NNN=1
      CALL START
C
C             CHECK IF THIS IS A RESTART
      IF(NCONT .EQ. 0)GOTO 15
      CALL WRTAPE (2)
      NPRINT=(NCYCLE-MOD(NCYCLE,NDELP))+NDELP
      GOTO 49
C        SET CYCLE NUMBER = 0
   15 NCYCLE=0
      TIME=0.0
      CINES=0.0
      TDAMP=0.0
      CINEP=0.0
      ENS=0.0
      D1=0.0
      D2=0.0
      D3=0.0
C          SET SYMMETRY BOUNDARY CONDITIONS FOR EDGE1,EDGE2,EDGE3
      IF(IBCE2 .NE. 2)GOTO 17
      DO 16 M=2,M1
      Y1(M,NN)=Y1(M,NR)
      Y2(M,NN)=-Y2(M,NR)+2.0*Y2(M,NS)
      Y3(M,NN)=Y3(M,NR)
```

```
   16 CONTINUE
   17 DO 20 N=1,NN
      Y1(1,N)=-Y1(3,N)
      Y2(1,N)= Y2(3,N)
      Y3(1,N)= Y3(3,N)
      IF(IBCE3 .NE. 2)GOTO 20
      Y1(MM,N)=-Y1(MR,N)
      Y2(MM,N)= Y2(MR,N)
      Y3(MM,N)= Y3(MR,N)
   20 CONTINUE
C
C     SET INITIAL DISPLACEMENT,PRESSURE,STRAIN AND STRESS = 0
      DO 28 M=1,MM
      DO 28 N=1,NN
      U1(M,N)=0.0
      U2(M,N)=0.0
      U3(M,N)=0.0
      P(M,N)=0.0
      EPSL1(M,N)=0.0
      EPSL2(M,N)=0.0
      GAMMAL(M,N)=0.0
      EPSU1(M,N)=0.0
      EPSU2(M,N)=0.0
      GAMMAU(M,N)=0.0
      DO 28 K=1,KJMAX
      SIG1(M,N,K)=0.
      SIG2(M,N,K)=0.
      TAU(M,N,K)=0.
   28 CONTINUE
C
      IF(LOAD) 29,30,29
   29 CALL PRESS
   30 DELGAM=DELSQ/GAMZ
C         WRITE INITIAL CARTESIAN COORDINATES, PRESSURE
      WRITE(6,300)
      DO 46 M=2,M1
   46 WRITE(6,400)M,(N,Y1(M,N),Y2(M,N),Y3(M,N),P(M,N),N=1,N1)
C
      CALL DGEOM
      CALL STRAIN
      IF(LOAD) 31,31,43
C
   31 CALL INVEL
C
      DO 35 M=2,MS
      DO 35 N=2,NS
      U1(M,N)=DELTAT*U1(M,N)
      U2(M,N)=DELTAT*U2(M,N)
      U3(M,N)=DELTAT*U3(M,N)
   35 CONTINUE
      CALL BOUNDU
      CALL KINET
      CINES=2.0*CINET
      TNRG=CINES
C
      IF(LOAD) 42,45,43
C
   42 CALL PWORK
C
   43 DO 44 M=2,MS
```

```
      DO 44 N=2,NS
      U1(M,N)=U1(M,N)-P(M,N)*SN1(M,N)*TEMP(M,N)
      U2(M,N)=U2(M,N)-P(M,N)*SN2(M,N)*TEMP(M,N)
      U3(M,N)=U3(M,N)-P(M,N)*SN3(M,N)*TEMP(M,N)
   44 CONTINUE
      CALL BOUNDU
      CALL PWORK
      ENR=ENS
      CALL KINET
      TNRG=CINET
   45 IF(NCYCM(1) .EQ. 0)NNN=2
   49 CALL PDATA (1)
C
C        END INITIALIZATION
   50 NCYCLE=NCYCLE+1
      TIME=TIME+DELTAT
C       CHECK FOR FINAL STEP
      IF(NCYCLE-MAXC) 60,60,70
C          CHECK IF CALL PRESS IS NEEDED
   60 IF(LPRESS-NCYCLE) 64,62,62
   62 CALL PRESS
   64 CALL POSITN
      CALL DGEOM
      CALL STRAIN
      CALL MOTION
      CALL PDATA (2)
      CALL DAMP
C             CHECK FOR RESTART DUMP
      IF(NCYCLE-NRITE) 50,66,50
   66 CALL WRTAPE (1)
      NRITE=NRITE+NRITE
      CALL PDATA (3)
      GOTO 50
   70 IF(NCYCLE .LT. 25)CALL EXIT
      CALL PDATA (3)
      CALL PDATA (4)
      CALL EXIT
C
  300 FORMAT(1H1,32X,29HINITIAL CARTESIAN COORDINATES,32X,8HPRESSURE/3X,
     11HM,3X,1HN,9X,7HY1(M,N),18X,7HY2(M,N),18X,7HY3(M,N),20X,6HP(M,N))
  400 FORMAT(2I4,4(2X,E23.16)/(I8,4(2X,E23.16)))
      END
```

```
      SUBROUTINE START
      DIMENSION TITLE(8)
      COMMON (USE MAIN)
      READ(5,100) TITLE
      READ(5,105) MESH,NMESH,LAYER,YLDFAC
      READ(5,105) MAXC,NCONT,NRITE,DELTAT
      READ(5,110) IBCE1,IBCE2,IBCE3,IBCE4
      READ(5,115) LOAD,LPRESS,MDAMP,DAMPF,DFACT
      READ(5,120) E,FNU,SIGZ,RHO,THICKN,NSFL,ISR
      IF(NSFL .EQ. 1 .AND. ISR .EQ. 0)GOTO 700
      IF(NSFL .EQ. 0 )GOTO 700
      READ (5,125) (SSIG(J),SEPS(J),DSR(J),PSR(J),J=1,NSFL)
  700 IF(NSFL .LT. 1) ISR=-1
      IF(NSFL .LT. 1) NSFL=1
      READ(5,110) NPRINT,(JCHK(J),J=1,3)
      READ(5,110) NUMCY,(NCYCH(J),J=1,NUMCY)
      READ(5,110) NLPRIN,(JCYNLP(J),J=1,NLPRIN)
      READ(5,110) N3D,(NC3DP(J),J=1,N3D)
      SSIG(1)=SIGZ
      SEPS(1)=SIGZ/E
      KJMAX=LAYER*NSFL
      SE(1)=E
      SIGZSQ(1)=SSIG(1)**2
      DO 795 J=1,NSFL
      IF(ISR .LT. 1)GOTO 794
      IF(DSR(J) .GT. 0.0 .AND. PSR(J) .GT. 0.0)GOTO 793
  791 WRITE(6,792)
  792 FORMAT(//46H ERROR IN STRAIN HARDENING OR STRAIN RATE DATA  )
      CALL EXIT
  793 PSR(J) = 1.0/PSR(J)
  794 IF(J .EQ. 1) GOTO 795
      IF(SEPS(J).LE.SEPS(J-1)) GOTO 791
      SE(J)=(SSIG(J)-SSIG(J-1))/(SEPS(J)-SEPS(J-1))
      WT(J-1)=(SE(J-1)-SE(J))/E
      SIGZSQ(J)=(E*SEPS(J))**2
  795 CONTINUE
      WT(NSFL)=SE(NSFL)/E
C       CHECK NUMBER OF MESH POINTS NEEDED FOR ETA1,ETA2 DIRECTIONS
      MM=MESH+2
      NN=NMESH+1
      N1=NN
C         CHECK BOUNDARY CONDTIONS FOR EDGE2
      IF(IBCE2 .NE. 2)GOTO 4
      NN=NMESH+2
      N1=NN-1
C
    4 M1=MM
      MB1=MM-2
      NB1=3
      NB2=NN-2
C         CHECK BOUNDARY CONDTIONS FOR EDGE3
      IF(IBCE3 .NE. 2)GOTO 5
      MM=MESH+3
      M1=MM-1
      MB1=MM-1
    5 MS=MM-1
      NS=NN-1
      MR=MM-2
      NR=NN-2
      IF(IBCE3 .EQ. 3) MB1=MS
```

```
      IF(IBCE4 .EQ. 3) NB1=2
      IF(IBCE2 .NE. 1) NB2=NS
C
      READ(5,130) ETAD1,ETAD2,NSTRN
      READ(5,15)(ETAG1(I),ETAG2(I),ANGLE(I),ANGLB(I),NETAG(I),I=1,NSTRN)
C
      CALL INGEOM
      QM=2.0+ETAD1/DETA1
      QN=1.0+ETAD2/DETA2
      MQ1=QM
      MQ2=MQ1+1
      NQ1=QN
      NQ2=NQ1+1
      QM1=QM-FLOAT(MQ1)
      QM2=FLOAT(MQ2)-QM
      QN1=QN-FLOAT(NQ1)
      QN2=FLOAT(NQ2)-QN
      DO 20 I=1,NSTRN
      PM(I)=2.0+ETAG1(I)/DETA1
      PN(I)=1.0+ETAG2(I)/DETA2
      MI1(I)=PM(I)
      MI2(I)=MI1(I)+1
      NI1(I)=PN(I)
      NI2(I)=NI1(I)+1
      PM1=MI1(I)
      PM2=MI2(I)
      PN1=NI1(I)
      PN2=NI2(I)
      DM1(I)=PM(I)-PM1
      DM2(I)=PM2-PM(I)
      DN1(I)=PN(I)-PN1
      DN2(I)=PN2-PN(I)
   20 CONTINUE
C        TIME INCREMENT BY VON NEUMAN
      C=SQRT(E/(RHO*(1.0-FNU**2)))
      DELM=2.0/(C*SQRT(1.0/DETA1**2+1.0/DETA2**2))
      ZAYER=LAYER
      FLAYER=(ZAYER/2.0)**2
      DETAN=.0833333333333333-(1.0/(48.0*FLAYER))
      DX4=1.0/DETA1**4
      DY4=1.0/DETA2**4
      T2=(1.0+15.0*FNU)/(8.0*DETA1**2*DETA2**2)
      DELB=1.0/(2.0*THICKN*C*SQRT(DETAN*(DX4+DY4+T2)))
      DELMIN=AMIN1(DELB,DELM)
      SUM=1.0
   25 RSUM=1.0/SUM
      SUM=SUM*10.0
      IF(RSUM .LT. DELMIN)GOTO 30
      GOTO 25
   30 DELMIN=AINT(DELMIN*SUM)/SUM
      DELIN=DELTAT
      IF(DELIN .GT. 0.0)DELMIN=AMIN1(DELIN,DELMIN)
      DELTAT=DELMIN
C              PROGRAM CONSTANTS
      H=0.5*THICKN
      GAMZ=THICKN*RHO
      NDELP=NPRINT
      MRITE=NCONT+MRITE
      DO 3 K=1,LAYER
      ZETA(K)=H*(1.0-(2.0*FLOAT(K)-1.0)/ZAYER)
```

```
      ZETASQ(K) = ZETA(K)**2
    3 CONTINUE
      TA=H*(2.0/ZAYER)
      DELSQ=DELTAT**2
      G=.5*E/(1.0+FNU)
      GTWO=G*2.0
      PRAT=E/(1.0-FNU**2)
      RTD1=1.0/(2.0*DETA1)
      RTD2=1.0/(2.0*DETA2)
      RD11=1.0/DETA1**2
      RD22=1.0/DETA2**2
      RD12=0.25/(DETA1*DETA2)
      CA=DETA1*DETA2
      CB=TA*DETA1*DETA2/E
      TRD=2.0*RD12
C         DAMPING CONSTANTS
      C2=2.0*DELTAT*DAMPF/GAMZ
      C1=C2/(4.0+C2)
C
      WRITE(6,130)
      WRITE(6,140) TITLE
      WRITE(6,150) MESH,DETA1,NMESH,DETA2
      WRITE(6,300) DELB,DELM,DELIN,DELTAT
      WRITE(6,160) E,FNU,SIGZ,RHO,THICKN
      WRITE(6,170) NCONT,MAXC,NPRINT,NRITE
      WRITE(6,175) LAYER,NSTRN,LOAD,LPRESS
      WRITE(6,180) IBCE1,IBCE2,IBCE3,IBCE4
      WRITE(6,185) (JCHK(I),I=1,3)
      WRITE(6,190) (NCYCH(I),I=1,NUMCY)
      WRITE(6,500) (JCYNLP(J),J=1,NLPRIN)
      WRITE(6,195) (NC3DP(I),I=1,N3D)
      IF(ISR .EQ. -1) WRITE(6,400)
      IF(NSFL .EQ. 1 .AND. ISR .EQ. 1) WRITE(6,405)
      IF(NSFL .GT. 1 .AND. ISR .EQ. 0) WRITE(6,410)
      IF(NSFL .EQ. 1 .AND. ISR .EQ. 0) WRITE(6,415)
      IF(NSFL .GT. 1 .AND. ISR .EQ. 1) WRITE(6,420)
      IF(NSFL.GT.1)WRITE(6,820)NSFL
      WRITE(6,821)(J,SSIG(J),SEPS(J),DSR(J),PSR(J),J=1,NSFL)
      WRITE(6,110)
      TIME=DELTAT*FLOAT(MDAMP)
      WRITE(6,200) MDAMP,TIME,DAMPF,DFACT
      RETURN
   15 FORMAT(4E10.4,I5)
  100 FORMAT(8A10)
  105 FORMAT(3I5,E12.6)
  110 FORMAT(16I5)
  115 FORMAT(3I5,2E12.6)
  120 FORMAT(5E12.6,2I5)
  125 FORMAT(4E15.7)
  130 FORMAT(2E10.4,I5)
C
  140 FORMAT(1H1,53X,15HBRL REPSIL CODE//24X,8A10/)
  150 FORMAT(38X,I4,29H MESHES IN THE ETA1 DIRECTION,3X,8H(DETA1 =E12.6,
     11H)/38X,I4,29H MESHES IN THE ETA2 DIRECTION,3X,8H(DETA2 =E12.6,1H)
     2/)
  160 FORMAT(57X,17HYOUNG'S MODULUS =E12.6
     1      /32X,17HPOISSON'S RATIO =E12.6,10X,17HYIELD STRESS    =E12.6
     2      /32X,17HMASS DENSITY    =E12.6,10X,17HTHICKNESS       =E12.6
     3/)
  170 FORMAT(55X,18HSTART AT TIME STEPI5/55X,18HFINAL  TIME  STEP ,I5/
```

```
     1       55X,21HSURFACE STRAINS EVERYI5,10H TIME STEP/                 STAR 181
     2       55X,21HRESTART  WRITE  EVERYI5,10H TIME STEP/)                STAR 182
 175 FORMAT(43X,7HLAYER =I5,18X,8HNSTRN  =I5/                              STAR 183
     1       43X,7HLOAD  =I5,18X,8HLPRESS =I5/)                            STAR 184
 180 FORMAT(54X,19HBOUNDARY CONDITIONS/47X,31H1/2/3 = CLAMPED/SYMMETRY/STAR 185
    1HINGED/57X,7HEDGE1 =I4/57X,7HEDGE2 =I4/57X,7HEDGE3 =I4/               STAR 186
     2       57X,7HEDGE4 =I4/)                                             STAR 187
 185 FORMAT(50X,25HPRINT OPTION CONTROL CARD/52X,20H0/1 = NO PRINT/PRINSTAR 188
    1T/50X,I4,24H DISPLACEMENT INCREMENTS/                                 STAR 189
     2  50X,I4,32H CARTESIAN COORDINATES, PRESSURE/                        STAR 190
     3  50X,I4,33H SURFACE NORMAL VECTOR COMPONENTS/)                      STAR 191
 190 FORMAT(24X,45HPRINT INFORMATION AT THE FOLLOWING TIME STEPS/(24X,  STAR 192
    1(16I5)))                                                              STAR 193
 195 FORMAT(24X,38H3-D PLOTS FOR THE FOLLOWING TIME STEPS/(24X,(16I5)))STAR 194
 200 FORMAT(/46X,29HSTART DAMPING AFTER TIME STEPI5,5X,6HTIME =E10.4/   STAR 195
    140X,7HDAMPF =E10.4,16X,7HDFACT =E10.4)                                STAR 196
 300 FORMAT(47X,25HBENDING  TIME INCREMENT= E12.6/47X,25HMEMBRANE TIME  STAR 197
    1INCREMENT= E12.6/47X,25HINPUT    TIME INCREMENT= E12.6//44X,31HTIMSTAR 198
    2E INCREMENT USED BY REPSIL= E12.6/)                                   STAR 199
 400 FORMAT(/37X,32HCONSTITUTIVE RELATION    ELASTIC)                      STAR 200
 405 FORMAT(/24X,78HCONSTITUTIVE RELATION    ELASTOPLASTIC-NO WORK HARDSTAR 201
    1ENING-STRAIN RATE DEPENDENT)                                          STAR 202
 410 FORMAT(/24X,77HCONSTITUTIVE RELATION    ELASTOPLASTIC-WORK HARDENISTAR 203
    1NG-STRAIN RATE INDEPENDENT)                                           STAR 204
 415 FORMAT(/24X,80HCONSTITUTIVE RELATION    ELASTOPLASTIC-NO WORK HARDSTAR 205
    1ENING-STRAIN RATE INDEPENDENT)                                        STAR 206
 420 FORMAT(/37X,75HCONSTITUTIVE RELATION    ELASTOPLASTIC-WORK HARDENISTAR 207
    1NG-STRAIN RATE DEPENDENT)                                             STAR 208
 500 FORMAT(24X,49HPRINT L MATRIX (LMAT) AT THE FOLLOWING TIME STEPS/   STAR 209
    1(24X,16I5))                                                           STAR 210
 820 FORMAT(35X,31HSTRESS-STRAIN APPROXIMATION HAS,I3,10H SUBLAYERS)    STAR 211
 821 FORMAT(/46X,40HSTRESS-STRAIN AND STRAIN RATE PARAMETERS/           STAR 212
    130X,1HJ,9X,7HSSIG(J),8X,7HSEPS(J),14X,6HDSR(J),8X,8H1/PSR(J)/      STAR 213
    2(26X,I5,5X,1P2E15.7,5X,2E15.7))                                       STAR 214
     END                                                                   STAR 215
```

```
      SUBROUTINE INVEL
      COMMON (USE MAIN)
C       EVALUATE THE INITIAL VELOCITY AT TIME=0 FOR ALL MESH POINTS
      READ(5,100) MI,MF,NI,NF,VR,NV
      WRITE(6,200) MI,MF,NI,NF,VR
C
      DO 30 M=MI,MF
      DO 30 N=NI,NF
      U1(M,N)=-VR*SN1(M,N)
      U2(M,N)=-VR*SN2(M,N)
      U3(M,N)=-VR*SN3(M,N)
   30 CONTINUE
      IF(NV)50,50,40
   40 WRITE(6,300)
      DO 45 K=1,NV
      READ(5,500) M,N,V
      WRITE(6,400) M,N,V
      U1(M,N)=-V*SN1(M,N)
      U2(M,N)=-V*SN2(M,N)
      U3(M,N)=-V*SN3(M,N)
   45 CONTINUE
   50 RETURN
C
  100 FORMAT(4I5,E12.6,I5)
  200 FORMAT(1H1,31X,3H(M=,I3,1H,,I3,9H) AND (N=,I3,1H,,I3,30H) RECEIVE
     1FULL VELOCITY,(VR)= ,E12.6/)
  300 FORMAT(52X,27HOTHER VELOCITY DISTRIBUTION/57X,1HM,4X,1HN,8X,1HV/)
  400 FORMAT(54X,2I5,2X,E12.6)
  500 FORMAT(2I5,E12.6)
      END
```

```
      SUBROUTINE POSITN
      COMMON(USE MAIN)
      DO 50 M=1,MM
      DO 50 N=1,NN
      Y1(M,N)=Y1(M,N)+U1(M,N)
      Y2(M,N)=Y2(M,N)+U2(M,N)
      Y3(M,N)=Y3(M,N)+U3(M,N)
   50 CONTINUE
      IF(NCYCLE .NE. NCYCH(NNN))GOTO 75
C       WRITE DISPLACEMENT INCREMENTS
      IF(JCHK(1)) 63,63,55
   55 NCYC=NCYCLE-1
      WRITE (6,509) NCYC,NCYCLE
      WRITE (6,9915)
      DO 60 M=2,M1
   60 WRITE (6,9917) M,(N,U1(M,N),U2(M,N),U3(M,N),N=1,N1)
C        WRITE CARTESIAN COORDINATES, PRESSURE
   63 IF(JCHK(2)) 75,75,65
   65 WRITE (6,99991) NCYCLE,TIME
      WRITE (6,903) NCYCLE,NCYCLE,NCYCLE
      DO 70 M=2,M1
   70 WRITE (6,902) M,(N,Y1(M,N),Y2(M,N),Y3(M,N),P(M,N),N=1,N1)
   75 RETURN
C
  509  FORMAT(1H1,21X,36HDISPLACEMENT INCREMENTS BETWEEN T.S.,I4,4H AND,
     1I4,/)
 9915 FORMAT(1H ,5X,1HM,4X,1HN,10X,7HU1(M,N),18X,7HU2(M,N),18X,7HU3(M,N)
     1)
 9917 FORMAT(I6,I5,3(2X,E23.16)/(I11,3(2X,E23.16)))
99991 FORMAT(10H1TIME STEP,I5,6X,5H TIME,E16.8)
  903 FORMAT(36X,21HCARTESIAN COORDINATES,36X,8HPRESSURE/3X,1HM,3X,1HN,
     19X,7HY1(M,N,I4,1H),13X,7HY2(M,N,I4,1H),13X,7HY3(M,N,I4,1H)
     2,15X,6HP(M,N))
  902 FORMAT(2I4,4(2X,E23.16)/(I8,4(2X,E23.16)))
      END
```

```
      SUBROUTINE DGEOM
      COMMON (USE MAIN)
      STREN=0.0
      DO 90 M=2,M1
      DO 90 N=1,N1
      CALL GRAD (M,N)
      RN1=SN1(M,N)
      RN2=SN2(M,N)
      RN3=SN3(M,N)
      A11=Y11**2+Y12**2+Y13**2
      A22=Y21**2+Y22**2+Y23**2
      A12=Y11*Y21+Y12*Y22+Y13*Y23
      DA=A11*A22-A12**2
      RA=1.0/DA
      SRA=SQRT(DA)
      RRA=1.0/SRA
      IF(LPRESS .GE. NCYCLE) P(M,N)=SRA*P(M,N)
      IF(NCYCLE .EQ. 0)GOTO 86
      IF(M.EQ.MM .AND. IBCE3.EQ.1)GOTO 87
      IF(N.EQ.1  .AND. IBCE4.EQ.1)GOTO 87
      IF(N.EQ.NN .AND. IBCE2.EQ.1)GOTO 87
   86 SN1(M,N)=RRA*(Y12*Y23-Y13*Y22)
      SN2(M,N)=RRA*(Y13*Y21-Y11*Y23)
      SN3(M,N)=RRA*(Y11*Y22-Y12*Y21)
   87 B11=SN1(M,N)*Y111+SN2(M,N)*Y112+SN3(M,N)*Y113
      B12=SN1(M,N)*Y121+SN2(M,N)*Y122+SN3(M,N)*Y123
      B22=SN1(M,N)*Y221+SN2(M,N)*Y222+SN3(M,N)*Y223
      IF (NCYCLE.GT.0) GOTO 88
      TEMP(M,N)=DELGAM*RRA
      CALL ABINIT(M,N)
      GOTO 90
   88 AR11= RA*A22
      AR22= RA*A11
      AR12=-RA*A12
      BM11=AR11*B11+AR12*B12
      BM12=AR11*B12+AR12*B22
      BM21=AR12*B11+AR22*B12
      BM22=AR12*B12+AR22*B22
      BT=BM11+BM22
      YR11=AR11*Y11+AR12*Y21
      YR12=AR11*Y12+AR12*Y22
      YR13=AR11*Y13+AR12*Y23
      YR21=AR12*Y11+AR22*Y21
      YR22=AR12*Y12+AR22*Y22
      YR23=AR12*Y13+AR22*Y23
      CS111=YR11*Y111+YR12*Y112+YR13*Y113
      CS112=YR11*Y121+YR12*Y122+YR13*Y123
      CS122=YR11*Y221+YR12*Y222+YR13*Y223
      CS211=YR21*Y111+YR22*Y112+YR23*Y113
      CS212=YR21*Y121+YR22*Y122+YR23*Y123
      CS222=YR21*Y221+YR22*Y222+YR23*Y223
      DNL1=-RN1*U11-RN2*U12-RN3*U13
      DNL2=-RN1*U21-RN2*U22-RN3*U23
      DNR1=AR11*DNL1+AR12*DNL2
      DNR2=AR12*DNL1+AR22*DNL2
      SNN=RN1*SN1(M,N)+RN2*SN2(M,N)+RN3*SN3(M,N)
      DN=(DNL1*DNR1+DNL2*DNR2)/(1.0+SNN)
      DA11=Y11*U11+Y12*U12+Y13*U13-0.5*(U11**2+U12**2+U13**2)
      DA22=Y21*U21+Y22*U22+Y23*U23-0.5*(U21**2+U22**2+U23**2)
      DA12=0.5*(Y11*U21+Y21*U11+Y12*U22+Y22*U12+Y13*U23+Y23*U13
```

```
     1                                        -U11*U21-U12*U22-U13*U23)
      DB11=RN1*U111+RN2*U112+RN3*U113+CS111*DNL1+CS211*DNL2+B11*DN
      DB22=RN1*U221+RN2*U222+RN3*U223+CS122*DNL1+CS222*DNL2+B22*DN
      DB12=RN1*U121+RN2*U122+RN3*U123+CS112*DNL1+CS212*DNL2+B12*DN
      DO 89 K=1,LAYER
      DEPS1(K)=DA11-ZETA(K)*DB11
      DEPS2(K)=DA22-ZETA(K)*DB22
      DGAMMA(K)=DA12-ZETA(K)*DB12
      CALL STRESS (M,N,K)
   89 CONTINUE
      EPSL1(M,N)=EPSL1(M,N)+DA11+H*DB11
      EPSL2(M,N)=EPSL2(M,N)+DA22+H*DB22
      GAMMAL(M,N)=GAMMAL(M,N)+DA12+H*DB12
      EPSU1(M,N)=EPSU1(M,N)+DA11-H*DB11
      EPSU2(M,N)=EPSU2(M,N)+DA22-H*DB22
      GAMMAU(M,N)=GAMMAU(M,N)+DA12-H*DB12
      CALL RESULT (M,N)
   90 CONTINUE
      IF(NCYCLE .EQ. 0)GOTO 820
      CALL SYMTRY
  100 STREN=CB*STREN
      IF(IBCE2 .EQ. 2) STREN=2.0*STREN
C         WRITE LMATRIX
      IF(NCYCLE.NE.JCYNLP(NLP))GOTO 820
      NLP=NLP+1
      WRITE(6,811)NCYCLE,TIME
      DO 802 K=1,LAYER
      WRITE(6,812)K,(M,M=2,M1)
      DO 802 N=1,N1
  802 WRITE(6,813)N,(LMAT(M,N,K),M=2,M1)
C         WRITE SURFACE NORMAL VECTOR
  820 IF(NCYCLE .NE. NCYCM(NNN))GOTO 700
      IF(JCHK(35)) 700,700,800
  800 WRITE(6,900) NCYCLE,TIME
      WRITE(6,910)
      DO 810 M=2,M1
  810 WRITE(6,920) M,(N,SN1(M,N),SN2(M,N),SN3(M,N),N=1,N1)
  700 RETURN
C
  811 FORMAT(10H1TIME STEP,I5,6X,4HTIME,1PE16.7,1H.,10X,
     1  40HSUBDIVISIONS OF TIME INCREMENT IN STRESS/)
  812 FORMAT(//20X,9HLMAT(M,N,I2,1H)//9H   N   M= ,40I3/)
  813 FORMAT(I3,5X,40I3)
  900 FORMAT(10H1TIME STEP,I5,6X,5H TIME,E16.8)
  910 FORMAT(22X,32HSURFACE NORMAL VECTOR COMPONENTS/3X,1
     1HM,3X,1HN,9X,8HSN1(M,N),17X,8HSN2(M,N),17X,8HSN3(M,N))
  920 FORMAT(2I4,3(2X,E23.16)/(I8,3(2X,E23.16)))
      END
```

```
      SUBROUTINE GRAD (MD,ND)
      COMMON (USE MAIN)
      M=MD
      N=ND
      IF (M.EQ.MM) GOTO 3
      Y11= RTD1*(Y1(M+1,N)-Y1(M-1,N))
      Y12= RTD1*(Y2(M+1,N)-Y2(M-1,N))
      Y13= RTD1*(Y3(M+1,N)-Y3(M-1,N))
      U11= RTD1*(U1(M+1,N)-U1(M-1,N))
      U12= RTD1*(U2(M+1,N)-U2(M-1,N))
      U13= RTD1*(U3(M+1,N)-U3(M-1,N))
      Y111=RD11*(Y1(M+1,N)-2.0*Y1(M,N)+Y1(M-1,N))
      Y112=RD11*(Y2(M+1,N)-2.0*Y2(M,N)+Y2(M-1,N))
      Y113=RD11*(Y3(M+1,N)-2.0*Y3(M,N)+Y3(M-1,N))
      U111=RD11*(U1(M+1,N)-2.0*U1(M,N)+U1(M-1,N))
      U112=RD11*(U2(M+1,N)-2.0*U2(M,N)+U2(M-1,N))
      U113=RD11*(U3(M+1,N)-2.0*U3(M,N)+U3(M-1,N))
      IF (N.EQ.1)  GOTO 1
      IF (N.EQ.NN) GOTO 2
      Y121=RD12*(Y1(M+1,N+1)-Y1(M-1,N+1)-Y1(M+1,N-1)+Y1(M-1,N-1))
      Y122=RD12*(Y2(M+1,N+1)-Y2(M-1,N+1)-Y2(M+1,N-1)+Y2(M-1,N-1))
      Y123=RD12*(Y3(M+1,N+1)-Y3(M-1,N+1)-Y3(M+1,N-1)+Y3(M-1,N-1))
      U121=RD12*(U1(M+1,N+1)-U1(M-1,N+1)-U1(M+1,N-1)+U1(M-1,N-1))
      U122=RD12*(U2(M+1,N+1)-U2(M-1,N+1)-U2(M+1,N-1)+U2(M-1,N-1))
      U123=RD12*(U3(M+1,N+1)-U3(M-1,N+1)-U3(M+1,N-1)+U3(M-1,N-1))
      GOTO 4
    1 Y121=RD12*(Y1(M-1,N+2)-Y1(M+1,N+2)
     1      -4.0*(Y1(M-1,N+1)-Y1(M+1,N+1)))-3.0*RTD2*Y11
      Y122=RD12*(Y2(M-1,N+2)-Y2(M+1,N+2)
     1      -4.0*(Y2(M-1,N+1)-Y2(M+1,N+1)))-3.0*RTD2*Y12
      Y123=RD12*(Y3(M-1,N+2)-Y3(M+1,N+2)
     1      -4.0*(Y3(M-1,N+1)-Y3(M+1,N+1)))-3.0*RTD2*Y13
      U121=RD12*(U1(M-1,N+2)-U1(M+1,N+2)
     1      -4.0*(U1(M-1,N+1)-U1(M+1,N+1)))-3.0*RTD2*U11
      U122=RD12*(U2(M-1,N+2)-U2(M+1,N+2)
     1      -4.0*(U2(M-1,N+1)-U2(M+1,N+1)))-3.0*RTD2*U12
      U123=RD12*(U3(M-1,N+2)-U3(M+1,N+2)
     1      -4.0*(U3(M-1,N+1)-U3(M+1,N+1)))-3.0*RTD2*U13
      GOTO 4
    2 Y121=RD12*(Y1(M+1,N-2)-Y1(M-1,N-2)
     1      -4.0*(Y1(M+1,N-1)-Y1(M-1,N-1)))+3.0*RTD2*Y11
      Y122=RD12*(Y2(M+1,N-2)-Y2(M-1,N-2)
     1      -4.0*(Y2(M+1,N-1)-Y2(M-1,N-1)))+3.0*RTD2*Y12
      Y123=RD12*(Y3(M+1,N-2)-Y3(M-1,N-2)
     1      -4.0*(Y3(M+1,N-1)-Y3(M-1,N-1)))+3.0*RTD2*Y13
      U121=RD12*(U1(M+1,N-2)-U1(M-1,N-2)
     1      -4.0*(U1(M+1,N-1)-U1(M-1,N-1)))+3.0*RTD2*U11
      U122=RD12*(U2(M+1,N-2)-U2(M-1,N-2)
     1      -4.0*(U2(M+1,N-1)-U2(M-1,N-1)))+3.0*RTD2*U12
      U123=RD12*(U3(M+1,N-2)-U3(M-1,N-2)
     1      -4.0*(U3(M+1,N-1)-U3(M-1,N-1)))+3.0*RTD2*U13
      GOTO 4
    3 Y11= RTD1*(Y1(M-2,N)-4.0*Y1(M-1,N)+3.0*Y1(M,N))
      Y12= RTD1*(Y2(M-2,N)-4.0*Y2(M-1,N)+3.0*Y2(M,N))
      Y13= RTD1*(Y3(M-2,N)-4.0*Y3(M-1,N)+3.0*Y3(M,N))
      U11= RTD1*(U1(M-2,N)-4.0*U1(M-1,N)+3.0*U1(M,N))
      U12= RTD1*(U2(M-2,N)-4.0*U2(M-1,N)+3.0*U2(M,N))
      U13= RTD1*(U3(M-2,N)-4.0*U3(M-1,N)+3.0*U3(M,N))
      Y111=RD11*(2.0*Y1(M,N)-5.0*Y1(M-1,N)+4.0*Y1(M-2,N)-Y1(M-3,N))
      Y112=RD11*(2.0*Y2(M,N)-5.0*Y2(M-1,N)+4.0*Y2(M-2,N)-Y2(M-3,N))
```

```
      V113=RD11*(2.0*V3(M,N)-5.0*V3(M-1,N)+4.0*V3(M-2,N)-V3(M-3,N))
      U111=RD11*(2.0*U1(M,N)-5.0*U1(M-1,N)+4.0*U1(M-2,N)-U1(M-3,N))
      U112=RD11*(2.0*U2(M,N)-5.0*U2(M-1,N)+4.0*U2(M-2,N)-U2(M-3,N))
      U113=RD11*(2.0*U3(M,N)-5.0*U3(M-1,N)+4.0*U3(M-2,N)-U3(M-3,N))
    4 IF (N.EQ.1)  GOTO 5
      IF (N.EQ.NN) GOTO 6
      V21= RTD2*(V1(M,N+1)-V1(M,N-1))
      V22= RTD2*(V2(M,N+1)-V2(M,N-1))
      V23= RTD2*(V3(M,N+1)-V3(M,N-1))
      U21= RTD2*(U1(M,N+1)-U1(M,N-1))
      U22= RTD2*(U2(M,N+1)-U2(M,N-1))
      U23= RTD2*(U3(M,N+1)-U3(M,N-1))
      V221=RD22*(V1(M,N+1)-2.0*V1(M,N)+V1(M,N-1))
      V222=RD22*(V2(M,N+1)-2.0*V2(M,N)+V2(M,N-1))
      V223=RD22*(V3(M,N+1)-2.0*V3(M,N)+V3(M,N-1))
      U221=RD22*(U1(M,N+1)-2.0*U1(M,N)+U1(M,N-1))
      U222=RD22*(U2(M,N+1)-2.0*U2(M,N)+U2(M,N-1))
      U223=RD22*(U3(M,N+1)-2.0*U3(M,N)+U3(M,N-1))
      IF (M.NE.MM) GOTO 7
      V121=RD12*(V1(M-2,N+1)-V1(M-2,N-1)
     1     -4.0*(V1(M-1,N+1)-V1(M-1,N-1)))+3.0*RTD1*V21
      V122=RD12*(V2(M-2,N+1)-V2(M-2,N-1)
     1     -4.0*(V2(M-1,N+1)-V2(M-1,N-1)))+3.0*RTD1*V22
      V123=RD12*(V3(M-2,N+1)-V3(M-2,N-1)
     1     -4.0*(V3(M-1,N+1)-V3(M-1,N-1)))+3.0*RTD1*V23
      U121=RD12*(U1(M-2,N+1)-U1(M-2,N-1)
     1     -4.0*(U1(M-1,N+1)-U1(M-1,N-1)))+3.0*RTD1*U21
      U122=RD12*(U2(M-2,N+1)-U2(M-2,N-1)
     1     -4.0*(U2(M-1,N+1)-U2(M-1,N-1)))+3.0*RTD1*U22
      U123=RD12*(U3(M-2,N+1)-U3(M-2,N-1)
     1     -4.0*(U3(M-1,N+1)-U3(M-1,N-1)))+3.0*RTD1*U23
      GOTO 7
    5 V21=-RTD2*(V1(M,N+2)-4.0*V1(M,N+1)+3.0*V1(M,N))
      V22=-RTD2*(V2(M,N+2)-4.0*V2(M,N+1)+3.0*V2(M,N))
      V23=-RTD2*(V3(M,N+2)-4.0*V3(M,N+1)+3.0*V3(M,N))
      U21=-RTD2*(U1(M,N+2)-4.0*U1(M,N+1)+3.0*U1(M,N))
      U22=-RTD2*(U2(M,N+2)-4.0*U2(M,N+1)+3.0*U2(M,N))
      U23=-RTD2*(U3(M,N+2)-4.0*U3(M,N+1)+3.0*U3(M,N))
      V221=RD22*(2.0*V1(M,N)-5.0*V1(M,N+1)+4.0*V1(M,N+2)-V1(M,N+3))
      V222=RD22*(2.0*V2(M,N)-5.0*V2(M,N+1)+4.0*V2(M,N+2)-V2(M,N+3))
      V223=RD22*(2.0*V3(M,N)-5.0*V3(M,N+1)+4.0*V3(M,N+2)-V3(M,N+3))
      U221=RD22*(2.0*U1(M,N)-5.0*U1(M,N+1)+4.0*U1(M,N+2)-U1(M,N+3))
      U222=RD22*(2.0*U2(M,N)-5.0*U2(M,N+1)+4.0*U2(M,N+2)-U2(M,N+3))
      U223=RD22*(2.0*U3(M,N)-5.0*U3(M,N+1)+4.0*U3(M,N+2)-U3(M,N+3))
      IF (M.NE.MM) GOTO 7
      V121=-RD12*(V1(M-2,N+2)-4.0*(V1(M-1,N+2)+V1(M-2,N+1)
     1     -4.0*V1(M-1,N+1))-9.0*V1(M,N))+3.0*(RTD1*V21-RTD2*V11)
      V122=-RD12*(V2(M-2,N+2)-4.0*(V2(M-1,N+2)+V2(M-2,N+1)
     1     -4.0*V2(M-1,N+1))-9.0*V2(M,N))+3.0*(RTD1*V22-RTD2*V12)
      V123=-RD12*(V3(M-2,N+2)-4.0*(V3(M-1,N+2)+V3(M-2,N+1)
     1     -4.0*V3(M-1,N+1))-9.0*V3(M,N))+3.0*(RTD1*V23-RTD2*V13)
      U121=-RD12*(U1(M-2,N+2)-4.0*(U1(M-1,N+2)+U1(M-2,N+1)
     1     -4.0*U1(M-1,N+1))-9.0*U1(M,N))+3.0*(RTD1*U21-RTD2*U11)
      U122=-RD12*(U2(M-2,N+2)-4.0*(U2(M-1,N+2)+U2(M-2,N+1)
     1     -4.0*U2(M-1,N+1))-9.0*U2(M,N))+3.0*(RTD1*U22-RTD2*U12)
      U123=-RD12*(U3(M-2,N+2)-4.0*(U3(M-1,N+2)+U3(M-2,N+1)
     1     -4.0*U3(M-1,N+1))-9.0*U3(M,N))+3.0*(RTD1*U23-RTD2*U13)
      GOTO 7
    6 V21= RTD2*(V1(M,N-2)-4.0*V1(M,N-1)+3.0*V1(M,N))
      V22= RTD2*(V2(M,N-2)-4.0*V2(M,N-1)+3.0*V2(M,N))
```

```
      Y23= RTD2*(Y3(M,N-2)-4.0*Y3(M,N-1)+3.0*Y3(M,N))
      U21= RTD2*(U1(M,N-2)-4.0*U1(M,N-1)+3.0*U1(M,N))
      U22= RTD2*(U2(M,N-2)-4.0*U2(M,N-1)+3.0*U2(M,N))
      U23= RTD2*(U3(M,N-2)-4.0*U3(M,N-1)+3.0*U3(M,N))
      Y221=RD22*(2.0*Y1(M,N)-5.0*Y1(M,N-1)+4.0*Y1(M,N-2)-Y1(M,N-3))
      Y222=RD22*(2.0*Y2(M,N)-5.0*Y2(M,N-1)+4.0*Y2(M,N-2)-Y2(M,N-3))
      Y223=RD22*(2.0*Y3(M,N)-5.0*Y3(M,N-1)+4.0*Y3(M,N-2)-Y3(M,N-3))
      U221=RD22*(2.0*U1(M,N)-5.0*U1(M,N-1)+4.0*U1(M,N-2)-U1(M,N-3))
      U222=RD22*(2.0*U2(M,N)-5.0*U2(M,N-1)+4.0*U2(M,N-2)-U2(M,N-3))
      U223=RD22*(2.0*U3(M,N)-5.0*U3(M,N-1)+4.0*U3(M,N-2)-U3(M,N-3))
      IF (M.NE.MM) GOTO 7
      Y121= RD12*(Y1(M-2,N-2)-4.0*(Y1(M-1,N-2)+Y1(M-2,N-1)
     1      -4.0*Y1(M-1,N-1))-9.0*Y1(M,N))+3.0*(RTD1*Y21+RTD2*Y11)
      Y122= RD12*(Y2(M-2,N-2)-4.0*(Y2(M-1,N-2)+Y2(M-2,N-1)
     1      -4.0*Y2(M-1,N-1))-9.0*Y2(M,N))+3.0*(RTD1*Y22+RTD2*Y12)
      Y123= RD12*(Y3(M-2,N-2)-4.0*(Y3(M-1,N-2)+Y3(M-2,N-1)
     1      -4.0*Y3(M-1,N-1))-9.0*Y3(M,N))+3.0*(RTD1*Y23+RTD2*Y13)
      U121= RD12*(U1(M-2,N-2)-4.0*(U1(M-1,N-2)+U1(M-2,N-1)
     1      -4.0*U1(M-1,N-1))-9.0*U1(M,N))+3.0*(RTD1*U21+RTD2*U11)
      U122= RD12*(U2(M-2,N-2)-4.0*(U2(M-1,N-2)+U2(M-2,N-1)
     1      -4.0*U2(M-1,N-1))-9.0*U2(M,N))+3.0*(RTD1*U22+RTD2*U12)
      U123= RD12*(U3(M-2,N-2)-4.0*(U3(M-1,N-2)+U3(M-2,N-1)
     1      -4.0*U3(M-1,N-1))-9.0*U3(M,N))+3.0*(RTD1*U23+RTD2*U13)
    7 RETURN
      END
```

```
      SUBROUTINE STRESS(MD,ND,KD)
C       CONSTITUTIVE RELATION--LINEARLY ELASTIC, OR ELASTIC-(-PERFECTLY
C       PLASTIC OR -STRAIN HARDENING).  OPTIONAL STRAIN RATE DEPENDENCE
      COMMON(USE MAIN)
      M=MD
      N=ND
      K=KD
      SS11=0.0
      SS12=0.0
      SS21=0.0
      SS22=0.0
      LMNK=0
      KN=(K-1)*NSFL
      ZETAK=2.0*ZETA(K)
      G11 =A11-ZETAK*B11
      G12 =A12-ZETAK*B12
      G22 =A22-ZETAX*B22
      DG=G11*G22-G12**2
      SRG=SQRT(DG)
      RG=1.0/DG
      GR11 = RG * G22
      GR12=-RG*G12
      GR22 = RG * G11
      DEPS11 = GR11*DEPS1(K) + GR12*DGAMMA(K)
      DEPS12 = GR12*DEPS2(K) + GR11*DGAMMA(K)
      DEPS21 = GR12*DEPS1(K) + GR22*DGAMMA(K)
      DEPS22 = GR22*DEPS2(K) + GR12*DGAMMA(K)
      DSIG11 = PRAT*(DEPS11 + FNU*DEPS22)
      DSIG12 = GTWO*DEPS12
      DSIG21 = GTWO*DEPS21
      DSIG22 = PRAT*(DEPS22 + FNU*DEPS11)
      IF(ISR.GT.0)EPSDOT=SQRT(DEPS11*(DEPS11-DEPS22)+DEPS22**2
     1+3.0*DEPS12*DEPS21)/DELTAT
      DO 1 J=1,NSFL
      KJ=KN+J
      SIG11I=G11*SIG1(M,N,KJ)+G12*TAU(M,N,KJ)
      SIG12I=G12*SIG1(M,N,KJ)+G22*TAU(M,N,KJ)
      SIG21I=G12*SIG2(M,N,KJ)+G11*TAU(M,N,KJ)
      SIG22I=G22*SIG2(M,N,KJ)+G12*TAU(M,N,KJ)
      SIGYSQ=SIGZSQ(J)
      IF(ISR.GT.0)SIGYSQ=SIGYSQ*(1.0+(EPSDOT/DSR(J))**PSR(J))**2
      I=1
      LC=1
      SIG11L=SIG11I+DSIG11
      SIG12L=SIG12I+DSIG12
      SIG21L=SIG21I+DSIG21
      SIG22L=SIG22I+DSIG22
      IF(ISR.LT.0)GOTO 2
      PHIT=SIG11L*(SIG11L-SIG22L)+SIG22L**2+3.0*SIG12L*SIG21L-SIGYSQ
      IF(PHIT.LE.0.0)GOTO 2
      L=INT(YLDFAC*(SQRT((PHIT+SIGYSQ)/SIGYSQ)-1.0))+1
  100 SIG11=SIG11I
      SIG12=SIG12I
      SIG21=SIG21I
      SIG22=SIG22I
      IF(L.EQ.1)GOTO 3
      LC=1
      FLOATL=1.0/FLOAT(L)
      DSG11L = DSIG11*FLOATL
      DSG12L = DSIG12*FLOATL
```

```
      DSG21L = DSIG21*FLOATL
      DSG22L = DSIG22*FLOATL
  101 SIG11L= SIG11+DSG11L
      SIG12L= SIG12+DSG12L
      SIG21L= SIG21+DSG21L
      SIG22L= SIG22+DSG22L
      PHIT=SIG11L*(SIG11L-SIG22L)+SIG22L**2+3.0*SIG12L*SIG21L-SIGYSQ
      IF(PHIT .GT. 0.0)GOTO 3
C     ELASTIC
    2 SIG11 = SIG11L
      SIG12 = SIG12L
      SIG21 = SIG21L
      SIG22 = SIG22L
      GO TO 9
C     PLASTIC
    3 SIG11D = (2.0-FNU)*SIG11-(1.0-2.0*FNU)*SIG22
      SIG12D = 3.0*(1.0-FNU)*SIG12
      SIG21D = 3.0*(1.0-FNU)*SIG21
      SIG22D = (2.0-FNU)*SIG22-(1.0-2.0*FNU)*SIG11
      AA = SIG11D**2-SIG11D*SIG22D+SIG22D**2+3.0*SIG12D*SIG21D
      B = -(SIG11L*(2.0*SIG11D-SIG22D)+SIG22L*(2.0*SIG22D-SIG11D)+3.0*(S
     1IG12L*SIG21D+SIG21L*SIG12D))
      D=B**2-4.*AA*PHIT
      IF(L .GT. LMNK) LMNK=L
      IF(AA)8,16,4
    8 WRITE(6,10)
   10 FORMAT(1H ,4X,14HAA NEGATIVE AT)
      GOTO 12
    4 IF(D .LT. 0.0 .OR. B .GT. 0.0)GOTO 16
      TAMBDA=(-B-SQRT(D))/(2.*AA)
      SIG11 = SIG11L-TAMBDA*SIG11D
      SIG12 = SIG12L-TAMBDA*SIG12D
      SIG21 = SIG21L-TAMBDA*SIG21D
      SIG22 = SIG22L-TAMBDA*SIG22D
    9 LC=LC+1
      IF(LC-L)101,101,102
   16 L=L+1
      IF(L-100)100,100,103
  103 WRITE(6,104)
  104 FORMAT(1H ,4X,36HSTRESS CALCULATION UNSATISFACTORY AT)
   12 WRITE(6,105)NCYCLE,M,N,K,L,LC
  105 FORMAT(1H ,9X,9HTIME STEP,I4,5X,2HM=,I2,5X,2HN=,I2,5X,2HK=,I2,5X,2
     1HL=,I3,5X,3HLC=,I3)
      WRITE(6,106) EPSL1(M,N),EPSL2(M,N),GAMMAL(M,N),EPSU1(M,N),
     1EPSU2(M,N),GAMMAU(M,N),DEPS1(K),DEPS2(K),DGAMMA(K),
     2 SIG1(M,N,KJ),SIG2(M,N,KJ),TAU(M,N,KJ),DSIG11,
     2DSIG12,DSIG21,DSIG22,SIG11,SIG12,SIG21,SIG22,SIG11L,SIG12L,SIG21L,
     3SIG22L,SIG11D,SIG12D,SIG21D,SIG22D,AA,B,PHIT
  106 FORMAT(1H ,9X,7HEPSL1 =E15.8,3X,7HEPSL2 =E15.8,3X,8HGAMMAL =E15.8,
     1/1H ,9X,7HEPSU1 =E15.8,3X,7HEPSU2 =E15.8,3X,8HGAMMAU =E15.8,
     1  /1H ,9X,7HDEPS1 =,E15.8,3X,7HDEPS2 =,E15.8,3X,7HDGAMMA=,E15.8,/1
     2H ,9X,7HSIG1   =,E15.8,3X,7HSIG2   =,E15.8,3X,7HTAU    =,E15.8,/1H ,9
     3X,7HDSIG11=,E15.8,3X,7HDSIG12=,E15.8,3X,7HDSIG21=,E15.8,3X,7HDSIG2
     42=,E15.8,/1H ,9X,7HSIG11 =,E15.8,3X,7HSIG12 =,E15.8,3X,7HSIG21 =,E
     515.8,3X,7HSIG22 =,E15.8,/1H ,9X,7HSIG11L=,E15.8,3X,7HSIG12L=,E15.8
     6,3X,7HSIG21L=,E15.8,3X,7HSIG22L=,E15.8,/1H ,9X,7HSIG11D=,E15.8,3X,
     77HSIG12D=,E15.8,3X,7HSIG21D=,E15.8,3X,7HSIG22D=,E15.8,/1H ,9X,7HAA
     8    =,E15.8,3X,7HB      =,E15.8,3X,7HPHIT  =,E15.8)
      NC3DP(NN3D)=NCYCLE
      CALL PDATA (2)
```

```
      CALL PDATA (3)
      CALL PDATA (4)
      CALL EXIT
  102 SS11=SS11+SIG11*WT(J)
      SS12=SS12+SIG12*WT(J)
      SS21=SS21+SIG21*WT(J)
      SS22=SS22+SIG22*WT(J)
      SIG1(M,N,KJ)=GR11*SIG11+GR12*SIG12
      SIG2(M,N,KJ)=GR22*SIG22+GR12*SIG21
      TAU (M,N,KJ)=GR12*SIG22+GR11*SIG21
  803 CONTINUE
      IF(M .EQ. 1 .OR. M .EQ. M1)SRG=0.5*SRG
      IF(N .EQ. 1 .OR. N .EQ. N1)SRG=0.5*SRG
      STREN=STREN+((SS11+SS22)**2-(1.+FNU)*2.*(SS11*SS22-SS12*SS21))*SRG
      SS1MN(K)=GR11*SS11+GR12*SS12
      SS2MN(K)=GR22*SS22+GR12*SS21
      STMN(K)=GR12*SS22+GR11*SS21
      LMAT(M,N,K) = LMNK
      RETURN
      END
```

```
      SUBROUTINE RESULT(MD,ND)
      COMMON (USE MAIN)
C     THE CALCULATION OF THE STRESS AND MOMENT RESULTANTS FOLLOW
      M=MD
      N=ND
      TB=TA*SRA
      SUMA11 = SS1MN(1)
      SUMA22 = SS2MN(1)
      SUMA12 =  STMN(1)
      SUMB11 = SS1MN(1)*ZETA(1)
      SUMB22 = SS2MN(1)*ZETA(1)
      SUMB12 =  STMN(1)*ZETA(1)
      SUMC11 = SS1MN(1)*ZETASQ(1)
      SUMC22 = SS2MN(1)*ZETASQ(1)
      SUMC12 =  STMN(1)*ZETASQ(1)
      IF(LAYER .EQ. 1)GOTO 5
      DO 4 K=2,LAYER
      SUMA11 = SUMA11 + SS1MN(K)
      SUMA22 = SUMA22 + SS2MN(K)
      SUMA12 = SUMA12 +  STMN(K)
      SUMB11 = SUMB11 + SS1MN(K)*ZETA(K)
      SUMB22 = SUMB22 + SS2MN(K)*ZETA(K)
      SUMB12 = SUMB12 +  STMN(K)*ZETA(K)
      SUMC11 = SUMC11 + SS1MN(K)*ZETASQ(K)
      SUMC22 = SUMC22 + SS2MN(K)*ZETASQ(K)
    4 SUMC12 = SUMC12 +  STMN(K)*ZETASQ(K)
    5 CONTINUE
      Q11=SUMA11-BT*SUMB11
      Q22=SUMA22-BT*SUMB22
      Q12=SUMA12-BT*SUMB12
      F11=SUMB11-(BT+BM11)*SUMC11-BM12*SUMC12
      F22=SUMB22-(BT+BM22)*SUMC22-BM21*SUMC12
      F12=SUMB12-1.5*BT*SUMC12-0.5*(BM21*SUMC11+BM12*SUMC22)
      IF(IBCE4 .NE. 3)GOTO 1
      IF(N .NE. 1)GOTO 1
      F22=0.0
    1 IF(IBCE2 .NE. 3)GOTO 2
      IF(N .NE. NN)GOTO 2
      F22=0.0
    2 IF(IBCE3 .NE. 3)GOTO 3
      IF(M .NE. MM)GOTO 3
      F11=0.0
    3 CSM1=CS111*F11+CS122*F22+2.0*CS112*F12
      CSM2=CS211*F11+CS222*F22+2.0*CS212*F12
      FNT11=Q11*Y11+Q12*Y21+CSM1*SN1(M,N)
      FNT12=Q11*Y12+Q12*Y22+CSM1*SN2(M,N)
      FNT13=Q11*Y13+Q12*Y23+CSM1*SN3(M,N)
      FNT21=Q12*Y11+Q22*Y21+CSM2*SN1(M,N)
      FNT22=Q12*Y12+Q22*Y22+CSM2*SN2(M,N)
      FNT23=Q12*Y13+Q22*Y23+CSM2*SN3(M,N)
      FM11(M,N) = TB*F11
      FM22(M,N) = TB*F22
      FM12(M,N) = TB*F12
      FN11(M,N) = TB*FNT11
      FN21(M,N) = TB*FNT21
      FN13(M,N) = TB*FNT13
      FN12(M,N) = TB*FNT12
      FN22(M,N) = TB*FNT22
      FN23(M,N) = TB*FNT23
      RETURN
```

END RESU 61

```
      SUBROUTINE MOTION
      COMMON(USE MAIN)
C
      VF1=0.0
      VF2=0.0
      VF3=0.0
      IF(LOAD) 10,30,10
   10 ENS=0.0
      CALL PWORK
C
   30 DO 130 M=2,MS
      DO 130 N=2,NS
      VM1=RD11*(FM11(M+1,N)*SN1(M+1,N)-2.0*FM11(M,N)*SN1(M,N)
     1   +FM11(M-1,N)*SN1(M-1,N))+    TRD  *(FM12(M+1,N+1)*SN1
     2   (M+1,N+1)-FM12(M+1,N-1)*SN1(M+1,N-1)-FM12(M-1,N+1)*SN1(M-1,N+1
     3   )+FM12(M-1,N-1)*SN1(M-1,N-1))+RD22*(FM22(M,N+1)*SN1(M,N+1)
     4   -2.0*FM22(M,N)*SN1(M,N)+FM22(M,N-1)*SN1(M,N-1))
      VM2=RD11*(FM11(M+1,N)*SN2(M+1,N)-2.0*FM11(M,N)*SN2(M,N)+FM11(M-1,N
     1  )*SN2(M-1,N))+    TRD  * FM12(M+1,N+1)*SN2(M+1,N+1)-FM12(M+1,N-1
     2   )*SN2(M+1,N-1)-FM12(M-1,N+1)*SN2(M-1,N+1)+FM12(M-1,N-1) *SN2(M
     3   -1,N-1))+RD22*(FM22(M,N+1)*SN2(M,N+1)-2.0*FM22(M,N)*SN2(M,N)+F
     4   M22(M,N-1)*SN2(M,N-1))
      VM3=RD11*(FM11(M+1,N)*SN3(M+1,N)-2.0*FM11(M,N)*SN3(M,N)+FM11(M-1,N
     1   )*SN3(M-1,N))+    TRD  *(FM12(M+1,N+1)*SN3(M+1,N+1)-FM12(M+1,N-
     2   1)*SN3(M+1,N-1)-FM12(M-1,N+1)*SN3(M-1,N+1)+FM12(M-1,N-1)*SN3
     3  (M-1,N-1))+RD22*(FM22(M,N+1)*SN3(M,N+1)-2.0*FM22(M,N)*SN3(M,N)
     4   +FM22(M,N-1)*SN3(M,N-1))
      VN1=RTD1*(FN11(M+1,N)-FN11(M-1,N))+RTD2*(FN21(M,N+1)-FN21(M,N-1))
      VN2=RTD1*(FN12(M+1,N)-FN12(M-1,N))+RTD2*(FN22(M,N+1)-FN22(M,N-1))
      VN3=RTD1*(FN13(M+1,N)-FN13(M-1,N))+RTD2*(FN23(M,N+1)-FN23(M,N-1))
      IF(LOAD) 40,50,40
   40 VF1=-SN1(M,N)*P(M,N)
      VF2=-SN2(M,N)*P(M,N)
      VF3=-SN3(M,N)*P(M,N)
C
   50 U1R=U1(M,N)
      U2R=U2(M,N)
      U3R=U3(M,N)
      U1S=U1R+(VM1+VN1+VF1)*TEMP(M,N)
      U2S=U2R+(VM2+VN2+VF2)*TEMP(M,N)
      U3S=U3R+(VM3+VN3+VF3)*TEMP(M,N)
      IF(TDAMP .EQ. 0.0)GOTO 115
C                             VISCOUS DAMPING  C1
      U1S=U1S-(U1S+U1R)*C1
      U2S=U2S-(U2S+U2R)*C1
      U3S=U3S-(U3S+U3R)*C1
  115 U1(M,N)=U1S
      U2(M,N)=U2S
      U3(M,N)=U3S
  130 CONTINUE
      CALL BOUNDU
      CALL KINET
      IF(LOAD) 65,75,65
   65 CALL PWORK
      EN=0.5*CA*(ENS+ENR)
      ENR=ENS
      IF(IBCE2 .NE. 2)GOTO 74
      EN=2.0*EN
   74 TNRG=TNRG+EN
   75 PLAST=TNRG-CINET-STREN-TDAMP
```

```
C
      IF(NCYCLE .NE. NCYCH(NNN)) GOTO 140
      WRITE(6,99991) NCYCLE,TIME,CINET,STREN,PLAST,TNRG
      NNN=NNN+1
  140 RETURN
C
99991 FORMAT(//10H TIME STEP,I5,3X,5HTIME=,E16.8,3X,8HKINETIC=,E15.8,3X,
     18HELASTIC=,E15.8,3X,8HPLASTIC=,E15.8/14H TOTAL ENERGY=,E15.8)
      END
```

```
      SUBROUTINE WRTAPE (KEY)
C       WRITE + READ CONTINUATION RUN TAPE   (KEY=1, WRITE  KEY=2, READ)
      COMMON(USE MAIN)
      GOTO(10,20),KEY
   10 WRITE(1) NCYCLE,TIME,CINES,CINEP,TNRG,TDAMP,D1,D2,D3,ENR
      WRITE(1) ((U1(M,N),U2(M,N),U3(M,N),Y1(M,N),Y2(M,N),Y3(M,N),
     1EPSL1(M,N),EPSL2(M,N),GAMMAL(M,N),EPSU1(M,N),EPSU2(M,N),
     2GAMMAU(M,N),P(M,N),N=1,NN),M=1,MM)
      WRITE(1) (((SIG1(M,N,K),SIG2(M,N,K),TAU(M,N,K),
     1K=1,KJMAX),N=1,NN),M=1,MM)
      WRITE(1) ((SN1(M,N),SN2(M,N),SN3(M,N),TEMP(M,N),N=1,NN),M=1,MM)
      WRITE(1) (GI11(N),GI22(N),GI12(N),ASA(N),BSA(N),CSA(N),ASB(N),
     1BSB(N),CSB(N),N=1,NSTRN)
      WRITE(6,100) NCYCLE,TIME
      RETURN
C
   20 REWIND 1
   25 READ (1) NCYCLE,TIME,CINES,CINEP,TNRG,TDAMP,D1,D2,D3,ENR
      READ (1) ((U1(M,N),U2(M,N),U3(M,N),Y1(M,N),Y2(M,N),Y3(M,N),
     1EPSL1(M,N),EPSL2(M,N),GAMMAL(M,N),EPSU1(M,N),EPSU2(M,N),
     2GAMMAU(M,N),P(M,N),N=1,NN),M=1,MM)
      READ (1) (((SIG1(M,N,K),SIG2(M,N,K),TAU(M,N,K),
     1K=1,KJMAX),N=1,NN),M=1,MM)
      READ (1)((SN1(M,N),SN2(M,N),SN3(M,N),TEMP(M,N),N=1,NN),M=1,MM)
      READ (1) (GI11(N),GI22(N),GI12(N),ASA(N),BSA(N),CSA(N),ASB(N),
     1BSB(N),CSB(N),N=1,NSTRN)
      IF(NCYCLE.NE.NCONT)GOTO 25
      WRITE(6,200) NCYCLE,TIME
      RETURN
C
  100 FORMAT(//24H TAPE 1 WRITTEN, NCYCLE=I4,7H  TIME=E15.8)
  200 FORMAT(//39H1INFORMATION READ FROM TAPE 1 FOR CYCLEI4,7H  TIME=E15
     1.8)
      END
```

```
      SUBROUTINE STRAIN
C
C     PRINT STRAINS ON INNER OR OUTER SURFACE
C
      DIMENSION EPSANG(6),EPSANB(6)
      COMMON (USE MAIN)
      DATA PI/3.141592653589793/
      IF(NCYCLE .GT. 0)GOTO 25
C     ------------------- INITIAL ENTRY -----------------------------
      DO 20 I=1,NSTRN
      A11=DM2(I)*(DN2(I)*A111(I)+DN1(I)*A112(I))
     1    +DM1(I)*(DN2(I)*A113(I)+DN1(I)*A114(I))
      A12=DM2(I)*(DN2(I)*A121(I)+DN1(I)*A122(I))
     1    +DM1(I)*(DN2(I)*A123(I)+DN1(I)*A124(I))
      A22=DM2(I)*(DN2(I)*A221(I)+DN1(I)*A222(I))
     1    +DM1(I)*(DN2(I)*A223(I)+DN1(I)*A224(I))
      B11=DM2(I)*(DN2(I)*B111(I)+DN1(I)*B112(I))
     1    +DM1(I)*(DN2(I)*B113(I)+DN1(I)*B114(I))
      B12=DM2(I)*(DN2(I)*B121(I)+DN1(I)*B122(I))
     1    +DM1(I)*(DN2(I)*B123(I)+DN1(I)*B124(I))
      B22=DM2(I)*(DN2(I)*B221(I)+DN1(I)*B222(I))
     1    +DM1(I)*(DN2(I)*B223(I)+DN1(I)*B224(I))
      IF(NETAG(I).EQ.1)GOTO 10
      G11 = A11-2.0*H*B11
      G12 = A12-2.0*H*B12
      G22 = A22-2.0*H*B22
      GOTO 15
   10 G11 = A11+2.0*H*B11
      G12 = A12+2.0*H*B12
      G22 = A22+2.0*H*B22
   15 GI11(I)=1.0/G11
      GI22(I)=1.0/G22
      GR=SQRT(GI11(I)*GI22(I))
      DELTA=G12*GR
      GI12(I)=2.0*GR
      SRDEL=1.0/SQRT(1.0-DELTA**2)
      ANGEL=ANGLE(I)*PI/180.0
      SA=SRDEL*SIN(ANGEL)
      SB=COS(ANGEL)-DELTA*SA
      ASA(I)=2.0*SA**2
      BSA(I)=2.0*SB**2
      CSA(I)=2.0*SA*SB
      ANGEL=ANGLB(I)*PI/180.0
      SA=SRDEL*SIN(ANGEL)
      SB=COS(ANGEL)-DELTA*SA
      ASB(I)=2.0*SA**2
      BSB(I)=2.0*SB**2
      CSB(I)=2.0*SA*SB
   20 CONTINUE
      GOTO 71
C     ---------------------------------------------------------------
   25 LINK=1
C          CHECK FOR SURFACE STRAIN PRINT
      IF(NCYCLE-NPRINT) 40,30,30
   30 NPRINT=NPRINT+NDELP
      LINK=2
   40 DO 46 I=1,NSTRN
      I1=MI1(I)
      J1=NI1(I)
      I2=MI2(I)
```

```
      J2=NI2(I)
      IF(NETAG(I) .EQ. 1) GOTO 44
      EPSR1   =(DM2(I)*(DN2(I)*EPSU1(I1,J1)+DN1(I)*EPSU1(I1,J2))
     1      +DM1(I)*(DN2(I)*EPSU1(I2,J1)+DN1(I)*EPSU1(I2,J2)))*GI11(I)
      EPSR2   =(DM2(I)*(DN2(I)*EPSU2(I1,J1)+DN1(I)*EPSU2(I1,J2))
     1      +DM1(I)*(DN2(I)*EPSU2(I2,J1)+DN1(I)*EPSU2(I2,J2)))*GI22(I)
      GAMMAR   =(DM2(I)*(DN2(I)*GAMMAU(I1,J1)+DN1(I)*GAMMAU(I1,J2))
     1       +DM1(I)*(DN2(I)*GAMMAU(I2,J1)+DN1(I)*GAMMAU(I2,J2)))*GI12(I)
      GOTO 45
   44 EPSR1   =(DM2(I)*(DN2(I)*EPSL1(I1,J1)+DN1(I)*EPSL1(I1,J2))
     1      +DM1(I)*(DN2(I)*EPSL1(I2,J1)+DN1(I)*EPSL1(I2,J2)))*GI11(I)
      EPSR2   =(DM2(I)*(DN2(I)*EPSL2(I1,J1)+DN1(I)*EPSL2(I1,J2))
     1      +DM1(I)*(DN2(I)*EPSL2(I2,J1)+DN1(I)*EPSL2(I2,J2)))*GI22(I)
      GAMMAR   =(DM2(I)*(DN2(I)*GAMMAL(I1,J1)+DN1(I)*GAMMAL(I1,J2))
     1       +DM1(I)*(DN2(I)*GAMMAL(I2,J1)+DN1(I)*GAMMAL(I2,J2)))*GI12(I)
   45 EPSS1(I)=SQRT(1.0+2.0*EPSR1)-1.0
      EPSS2(I)=SQRT(1.0+2.0*EPSR2)-1.0
      EPSANG(I)=SQRT(1.0+BSA(I)*EPSR1+ASA(I)*EPSR2+CSA(I)*GAMMAR)-1.0
      EPSANB(I)=SQRT(1.0+BSB(I)*EPSR1+ASB(I)*EPSR2+CSB(I)*GAMMAR)-1.0
   46 CONTINUE
C          COMPONENTS OF VECTOR DISPLACEMENT
      D1=D1+(QM2*(QN2*U1(MQ1,NQ1)+QN1*U1(MQ1,NQ2))
     1         +QM1*(QN2*U1(MQ2,NQ1)+QN1*U1(MQ2,NQ2)))
      D2=D2+(QM2*(QN2*U2(MQ1,NQ1)+QN1*U2(MQ1,NQ2))
     1         +QM1*(QN2*U2(MQ2,NQ1)+QN1*U2(MQ2,NQ2)))
      D3=D3+(QM2*(QN2*U3(MQ1,NQ1)+QN1*U3(MQ1,NQ2))
     1         +QM1*(QN2*U3(MQ2,NQ1)+QN1*U3(MQ2,NQ2)))
      GOTO (71,50),LINK
   50 WRITE(6,60) NCYCLE,TIME
      DO 70 I=1,NSTRN
      ALFN=5HINNER
      IF(NETAG(I).EQ.1)GOTO 67
      ALFN=5HOUTER
   67 WRITE(6,65) ETAG1(I),ETAG2(I),PM(I),PN(I),ALFN,EPSS1(I),EPSS2(I),
     1ANGLE(I),EPSANG(I),ANGLB(I),EPSANB(I)
   70 CONTINUE
   71 RETURN
C
   60 FORMAT(//10H TIME STEP,I5,3X,5HTIME=,E16.8//14X,15HSURFACE STRAINS
     1,37X,19HSTRAIN GAGE READING//2X,4HETA1,4X,4HETA2,6X,1HM,7X,1HN,5X,
     24HFACE,8X,7HANGLE 0,10X,8HANGLE 90,6X,5HANGLE,18X,5HANGLE/)
   65 FORMAT(1H ,F7.3,1X,F7.3,2X,F7.3,1X,F7.3,2X,A5,1X,2(2X,E15.8),
     12(2X,F6.2,2X,E15.8))
      END
```

```
      SUBROUTINE BOUNDU
C        COMPUTATION OF U1,U2,U3 AT THE BOUNDARY
      COMMON (USE MAIN)
      IF(IBCE4 .EQ. 3)GOTO 121
      DO 120 M=2,MB1
      DUSN=(U1(M,2)-0.25*U1(M,3))*SN1(M,1)
     1     +(U2(M,2)-0.25*U2(M,3))*SN2(M,1)
     2     +(U3(M,2)-0.25*U3(M,3))*SN3(M,1)
      U1(M,2)=U1(M,2)-SN1(M,1)*DUSN
      U2(M,2)=U2(M,2)-SN2(M,1)*DUSN
  120 U3(M,2)=U3(M,2)-SN3(M,1)*DUSN
  121 IF(IBCE2 .NE. 1)GOTO 123
      DO 122 M=2,MB1
      DUSN=(U1(M,NS)-0.25*U1(M,NR))*SN1(M,NN)
     1     +(U2(M,NS)-0.25*U2(M,NR))*SN2(M,NN)
     2     +(U3(M,NS)-0.25*U3(M,NR))*SN3(M,NN)
      U1(M,NS)=U1(M,NS)-SN1(M,NN)*DUSN
      U2(M,NS)=U2(M,NS)-SN2(M,NN)*DUSN
  122 U3(M,NS)=U3(M,NS)-SN3(M,NN)*DUSN
  123 IF(IBCE3 .NE. 1)GOTO 50
      DO 124 N=NB1,NB2
      DUSN=(U1(MS,N)-0.25*U1(MR,N))*SN1(MM,N)
     1     +(U2(MS,N)-0.25*U2(MR,N))*SN2(MM,N)
     2     +(U3(MS,N)-0.25*U3(MR,N))*SN3(MM,N)
      U1(MS,N)=U1(MS,N)-SN1(MM,N)*DUSN
      U2(MS,N)=U2(MS,N)-SN2(MM,N)*DUSN
  124 U3(MS,N)=U3(MS,N)-SN3(MM,N)*DUSN
      IF(IBCE4 .NE. 1)GOTO 125
      DUSN1=(U1(MS,2)-0.25*U1(MS,3))*SN1(MS,1)
     1      +(U2(MS,2)-0.25*U2(MS,3))*SN2(MS,1)
     2      +(U3(MS,2)-0.25*U3(MS,3))*SN3(MS,1)
      DUSN2=(U1(MS,2)-0.25*U1(MR,2))*SN1(MM,2)
     1      +(U2(MS,2)-0.25*U2(MR,2))*SN2(MM,2)
     2      +(U3(MS,2)-0.25*U3(MR,2))*SN3(MM,2)
      U1(MS,2)=U1(MS,2)-0.5*(SN1(MS,1)*DUSN1+SN1(MM,2)*DUSN2)
      U2(MS,2)=U2(MS,2)-0.5*(SN2(MS,1)*DUSN1+SN2(MM,2)*DUSN2)
      U3(MS,2)=U3(MS,2)-0.5*(SN3(MS,1)*DUSN1+SN3(MM,2)*DUSN2)
  125 IF(IBCE2 .NE. 1)GOTO 50
      DUSN2=(U1(MS,NS)-0.25*U1(MR,NS))*SN1(MM,NS)
     1      +(U2(MS,NS)-0.25*U2(MR,NS))*SN2(MM,NS)
     2      +(U3(MS,NS)-0.25*U3(MR,NS))*SN3(MM,NS)
      DUSN1=(U1(MS,NS)-0.25*U1(MS,NR))*SN1(MS,NN)
     1      +(U2(MS,NS)-0.25*U2(MS,NR))*SN2(MS,NN)
     2      +(U3(MS,NS)-0.25*U3(MS,NR))*SN3(MS,NN)
      U1(MS,NS)=U1(MS,NS)-0.5*(SN1(MS,NN)*DUSN1+SN1(MM,NS)*DUSN2)
      U2(MS,NS)=U2(MS,NS)-0.5*(SN2(MS,NN)*DUSN1+SN2(MM,NS)*DUSN2)
      U3(MS,NS)=U3(MS,NS)-0.5*(SN3(MS,NN)*DUSN1+SN3(MM,NS)*DUSN2)
C       SET SYMMETRY BOUNDARY CONDITIONS FOR EDGE1,EDGE2,EDGE3
   50 IF(IBCE2 .NE. 2)GOTO 30
      DO 25 M=2,M1
      U1(M,NN)=U1(M,NR)
      U2(M,NN)=-U2(M,NR)
      U3(M,NN)=U3(M,NR)
      U2(M,NS)=0.0
   25 CONTINUE
   30 DO 5 N=1,NN
      U1(2,N)= 0.
      U1(1,N)=-U1(3,N)
      U2(1,N)= U2(3,N)
      U3(1,N)= U3(3,N)
```

```
      IF(IBCE3 .NE. 2)GOTO 5
      U1(NS,N)= 0.0
      U1(MM,N)=-U1(MR,N)
      U2(MM,N)= U2(MR,N)
      U3(MM,N)= U3(MR,N)
 5    CONTINUE
      RETURN
      END
```

```
      SUBROUTINE ABINIT(M,N)                                            ****   1
      COMMON (USE MAIN)                                                 ABIN   2
      DO 200 I=1,NSTRN                                                  ABIN   3
      IF(M.EQ.NI1(I) .AND. (N.EQ.NI1(I) .OR. N.EQ.NI2(I)))GOTO 205      ABIN   4
      IF(M.EQ.NI2(I) .AND. (N.EQ.NI1(I) .OR. N.EQ.NI2(I)))GOTO 220      ABIN   5
      GOTO 200                                                          ABIN   6
  205 IF(N .EQ. NI2(I))GOTO 210                                         ABIN   7
      A111(I)=A11                                                       ABIN   8
      A121(I)=A12                                                       ABIN   9
      A221(I)=A22                                                       ABIN  10
      B111(I)=B11                                                       ABIN  11
      B121(I)=B12                                                       ABIN  12
      B221(I)=B22                                                       ABIN  13
      GOTO 200                                                          ABIN  14
  210 A112(I)=A11                                                       ABIN  15
      A122(I)=A12                                                       ABIN  16
      A222(I)=A22                                                       ABIN  17
      B112(I)=B11                                                       ABIN  18
      B122(I)=B12                                                       ABIN  19
      B222(I)=B22                                                       ABIN  20
      GOTO 200                                                          ABIN  21
  220 IF(N .EQ. NI2(I))GOTO 225                                         ABIN  22
      A113(I)=A11                                                       ABIN  23
      A123(I)=A12                                                       ABIN  24
      A223(I)=A22                                                       ABIN  25
      B113(I)=B11                                                       ABIN  26
      B123(I)=B12                                                       ABIN  27
      B223(I)=B22                                                       ABIN  28
      GOTO 200                                                          ABIN  29
  225 A114(I)=A11                                                       ABIN  30
      A124(I)=A12                                                       ABIN  31
      A224(I)=A22                                                       ABIN  32
      B114(I)=B11                                                       ABIN  33
      B124(I)=B12                                                       ABIN  34
      B224(I)=B22                                                       ABIN  35
  200 CONTINUE                                                          ABIN  36
      RETURN                                                            ABIN  37
      END                                                               ABIN  38
```

```
      SUBROUTINE SYMTRY
      COMMON (USE MAIN)
      DO 10 N=1,N1
      SN1(1,N)=-SN1(3,N)
      SN2(1,N)= SN2(3,N)
      SN3(1,N)= SN3(3,N)
      FM11(1,N)= FM11(3,N)
      FM22(1,N)= FM22(3,N)
      FM12(1,N)=-FM12(3,N)
      FN11(1,N)= FN11(3,N)
      FN21(1,N)=-FN21(3,N)
      FN13(1,N)=-FN13(3,N)
      FN12(1,N)=-FN12(3,N)
      FN22(1,N)= FN22(3,N)
      FN23(1,N)= FN23(3,N)
   10 CONTINUE
      IF(IBCE3 .NE. 2)GOTO 30
      DO 20 N=1,N1
      SN1(MM,N)=-SN1(MR,N)
      SN2(MM,N)= SN2(MR,N)
      SN3(MM,N)= SN3(MR,N)
      FM11(MM,N)= FM11(MR,N)
      FM22(MM,N)= FM22(MR,N)
      FM12(MM,N)=-FM12(MR,N)
      FN11(MM,N)= FN11(MR,N)
      FN21(MM,N)=-FN21(MR,N)
      FN13(MM,N)=-FN13(MR,N)
      FN12(MM,N)=-FN12(MR,N)
      FN22(MM,N)= FN22(MR,N)
      FN23(MM,N)= FN23(MR,N)
   20 CONTINUE
   30 IF(IBCE2 .NE. 2)GOTO 50
      DO 40 M=1,MM
      SN1(M,NN)= SN1(M,NR)
      SN2(M,NN)=-SN2(M,NR)
      SN3(M,NN)= SN3(M,NR)
      FM11(M,NN)= FM11(M,NR)
      FM12(M,NN)=-FM12(M,NR)
      FM22(M,NN)= FM22(M,NR)
      FN11(M,NN)= FN11(M,NR)
      FN21(M,NN)=-FN21(M,NR)
      FN13(M,NN)= FN13(M,NR)
      FN12(M,NN)=-FN12(M,NR)
      FN22(M,NN)= FN22(M,NR)
      FN23(M,NN)=-FN23(M,NR)
   40 CONTINUE
   50 RETURN
      END
```

```
      SUBROUTINE KINET
      COMMON (USE MAIN)
      CINET=0.
      DO 20 M=2,MS
      CM=1.0
      IF(M.EQ.2 .OR. (IBCE3.EQ.2 .AND. M.EQ.MS))CM=0.5
      DO 20 N=2,NS
      CN=1.0
      IF(IBCE2 .EQ. 2 .AND. N .EQ. NS)CN=0.5
   20 CINET=CINET+(U1(M,N)**2+U2(M,N)**2+U3(M,N)**2)/TEMP(M,N)*CM*CN
      IF(IBCE2 .NE. 2)GOTO 25
      CINET=2.0*CINET
   25 CINER=CINES
      CINES=CA*CINET
      CINET=0.5*(CINES+CINER)
      RETURN
      END
```

```
      SUBROUTINE PWORK
      COMMON (USE MAIN)
      DO 20 M=2,MS
      CM=1.0
      IF(M.EQ.2 .OR. (IBCE3.EQ.2 .AND. M.EQ.MS))CM=0.5
      DO 20 N=2,NS
      CN=1.0
      IF(IBCE2 .EQ. 2 .AND. N .EQ. NS)CN=0.5
      DW=U1(M,N)*SN1(M,N)+U2(M,N)*SN2(M,N)+U3(M,N)*SN3(M,N)
      ENS=ENS-CM*CN*DW*P(M,N)
   20 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE DAMP
      COMMON (USE MAIN)
C
C           CHECK FOR START OF DAMPING
      IF(NCYCLE .LT. MDAMP)RETURN
      DO 15 M=1,MM
      DO 15 N=1,NN
      P(M,N)=0.0
   15 CONTINUE
      IF(CINES-CINER)20,20,40
   20 TDAMP=TDAMP+CINET
      IF(CINET+CINEP .LE. DFACT*TDAMP)GOTO 50
      DO 30 M=1,MM
      DO 30 N=1,NN
      U1(M,N)=0.0
      U2(M,N)=0.0
      U3(M,N)=0.0
   30 CONTINUE
      CINEP=CINET
      CINES1=CINER
      CINES=0.0
      NCYCLE=NCYCLE+1
      TIME=TIME+DELTAT
      CALL MOTION
      IF(CINES .LE. CINES1)GOTO 39
      CALL DESTEP
   39 CALL PDATA(2)
   40 TDAMP=TDAMP+C2*CINES
   45 RETURN
   50 WRITE(6,100) NCYCLE
      MAXC=NCYCLE
      NC3DP(NN3D)=NCYCLE
      CALL PDATA (2)
      GOTO 45
  100 FORMAT(1H1,10X,30HRUN SELF-TERMINATED TIME STEP,I5)
      END
```

```
      SUBROUTINE DESTEP
C        CHANGE DELTAT
      COMMON (USE MAIN)
      DELTAT=SQRT(CINES1/CINES)*DELTAT
      DSQOLD=DELSQ
      C1OLD=C1
      DELSQ=DELTAT**2
      C2=2.0*DELTAT*DAMPF/GAMZ
      C1=C2/(4.0+C2)
      DELR=DELSQ/DSQOLD
      DELS=DELR*(1.0-C1)/(1.0-C1OLD)
      CINES=CINES*DELS**2/DELR
      CINET=0.5*(CINES+CINER)
      PLAST=TNRG-CINET-STREN-TDAMP
      DO 10 M=2,M1
      DO 10 N=1,N1
      TEMP(M,N)=DELR*TEMP(M,N)
      U1(M,N)=DELS*U1(M,N)
      U2(M,N)=DELS*U2(M,N)
      U3(M,N)=DELS*U3(M,N)
   10 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE PDATA(LINK)
      DIMENSION DAT(20)
      COMMON (USE MAIN)
C        PDATA SELECTS AND WRITES DATA ON TAPE(NPLOT, FOR THE REPSIL
C       PLOTTING PROGRAM
C
      GOTO (10,40,50,60),LINK
   10 NN3D=1
      II=2*NSTRN+8
      IF(NCYCLE .EQ. 0)GOTO 15
   12 CALL SKIPFILE (NPLOT,1)
      READ(NPLOT) IFLAG
      IF(IFLAG .EQ. 99999)GOTO 14
      READ(NPLOT) IFLAG
      IF(IFLAG .NE. NCONT+1)GOTO 12
   14 CALL BACKFILE (NPLOT,1)
      GOTO 50
   15 WRITE(NPLOT) IBCE3,ETAD1,ETAD2,QM,QN,NSTRN
      WRITE(NPLOT) (ETAG1(I),ETAG2(I),PM(I),PN(I),NETAG(I),I=1,NSTRN)
      WRITE(NPLOT) NCYCLE,TIME,M1,N1,((Y1(M,N),Y2(M,N),Y3(M,N),M=1,M1),
     1N=1,N1)
C
      DO 25 I=1,II
      DAT(I)=0.0
   25 CONTINUE
      IF(LOAD) 30,30,35
   30 DAT(5)=TNRG
      DAT(6)=TNRG
      DAT(7)=TNRG
      DAT(8)=TNRG
   35 IFLAG=1
      WRITE(NPLOT) IFLAG
      WRITE(NPLOT) NCYCLE,(DAT(I),I=1,II)
      GOTO 100
C
   40 DAT(1)=TIME
      DAT(2)=D1
      DAT(3)=D2
      DAT(4)=D3
      DAT(5)=CINET
      DAT(6)=STREN+CINET
      DAT(7)=TNRG
      DAT(8)=DAT(6)+TDAMP
      J=9
      DO 45 I=1,NSTRN
      DAT(J)=EPSS1(I)
      DAT(J+1)=EPSS2(I)
      J=J+2
   45 CONTINUE
      IFLAG=1
      WRITE(NPLOT) IFLAG
      WRITE(NPLOT) NCYCLE,(DAT(I),I=1,II)
C           CHECK FOR 3D PLOT
      IF(NCYCLE .NE. NC3DP(NN3D))GOTO 100
      NN3D=NN3D+1
      IFLAG=2
      WRITE(NPLOT) IFLAG
      WRITE(NPLOT) NCYCLE,TIME,M1,N1,((Y1(M,N),Y2(M,N),Y3(M,N),M=1,M1),
     1N=1,N1)
      GOTO 100
```

```
C
   50 END FILE NPLOT
      GOTO 100
C
   60 IFLAG=99999
      WRITE(NPLOT) IFLAG
  100 RETURN
      END
```

```
      SUBROUTINE PRESS
      COMMON (USE MAIN)
      COMMON /IPR/ RSQ(23,34)
      DO 10 M=2,MS
      DO 10 N=2,NS
      TI=(SQRT(RSQ(M,N)+225.0)-15.0)/144000.0
      P(M,N)=0.0
      IF(TIME .LT. TI)GOTO 10
      P(M,N)=5504737.5*EXP(-13000.0*(TIME-TI))/(RSQ(M,N)+225.0)
   10 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE INGEOM
C                     FLAT PLATE
      COMMON (USE MAIN)
      COMMON /IPR/ RSQ(23,34)
      REAL LENGTH
C
      READ (5,100) LENGTH,WIDTH
      DETA1=WIDTH/FLOAT(MESH)
      DETA2=LENGTH/FLOAT(NMESH)
C
      DO 10 M=2,M1
      DO 10 N=1,N1
      Y1(M,N)=FLOAT(M-2)*DETA1
      Y2(M,N)=FLOAT(N-1)*DETA2
      Y3(M,N)=0.0
      RSQ(M,N)=Y1(M,N)**2+(Y2(M,N)-LENGTH)**2
   10 CONTINUE
      RETURN
C
  100 FORMAT(2E10.4)
      END
```

```
      SUBROUTINE INGEOM
C              CYLINDER
      COMMON (USE MAIN)
      REAL LENGTH
      DATA DTOR/.1745329251994329E-01/
C
      READ (5,100) LENGTH,RADIUS,THETA
      DETA1=THETA*RADIUS*DTOR/FLOAT(MESH)
      DETA2=LENGTH/FLOAT(NMESH)
      DO 10 M=2,M1
      ETA1=FLOAT(M-2)*DETA1
      DO 10 N=1,N1
      ETA2=FLOAT(N-1)*DETA2
      Y1(M,N)=RADIUS*SIN(ETA1/RADIUS)
      Y2(M,N)=ETA2
      Y3(M,N)=RADIUS*COS(ETA1/RADIUS)
   10 CONTINUE
C
      ETAD1=ETAD1*RADIUS*DTOR
      DO 20 I=1,NSTRN
      ETAG1(I)=ETAG1(I)*RADIUS*DTOR
   20 CONTINUE
      RETURN
C
  100 FORMAT(3E12.6)
      END
```

```
      SUBROUTINE INGEOM
C                     CONICAL SHELL
      COMMON (USE MAIN)
      REAL LENGTH
      DATA DTOR/.1745329251994329E-01/
C
      READ(5,100) LENGTH,RADI,RADF,THETA,MASH
      ALFA=ATAN((RADF-RADI)/LENGTH)
      SINALF=SIN(ALPA)
      CSCALF=1.0/SINALF
      COTALF=LENGTH/(RADF-RADI)
      ETA1F=RADI*THETA*DTOR
      IF(MASH .EQ. 0)GOTO 1
      ETA2I=0.0
      ETA2F=RADI*CSCALF*ALOG(RADF/RADI)
      GOTO 2
    1 ETA2I=RADI*CSCALF
      ETA2F=RADF*CSCALF
    2 DETA1=ETA1F/FLOAT(MESH)
      DETA2=(ETA2F-ETA2I)/FLOAT(NMESH)
      DCHI1=DETA1/RADI
      DCHI2=DETA2*SINALF/RADI
      CHI2I=ETA2I*SINALF/RADI
      MU=M1
      IF(THETA .LT. 180.0)GOTO 5
      MM=1+MS/2
      MM1=MM+1
      MU=MM
      IF(2*MM .EQ. MS+2) MU=1+MM/2
    5 DO 10 N=1,N1
      CHI2=FLOAT(N-1)*DCHI2+CHI2I
      FINK=CHI2
      IF(MASH .GT. 0) FINK=EXP(CHI2)
      DO 10 M=2,MU
      CHI1=FLOAT(M-2)*DCHI1
      Y1(M,N)=RADI*SIN(CHI1)*FINK
      Y2(M,N)=RADI*COTALF*(FINK-1.0)
      Y3(M,N)=RADI*COS(CHI1)*FINK
   10 CONTINUE
      IF(THETA .LT. 180.0)GOTO 50
      IF(2*MM .LT. MS+2)GOTO 30
      MU1=MU+1
      IF(2*MU .EQ. MM+2) MU1=MU
      DO 20 N=1,N1
      DO 20 M=MU1,MM
      MK=MM+2-M
      Y1(M,N)=Y3(MK,N)
      Y2(M,N)=Y2(MK,N)
   20 Y3(M,N)=Y1(MK,N)
   30 DO 40 N=1,N1
      DO 40 M=MM1,MS
      MK=MS+2-M
      Y1(M,N)=Y1(MK,N)
      Y2(M,N)=Y2(MK,N)
   40 Y3(M,N)=-Y3(MK,N)
C
      CSCRAD=CSCALF*RADI
      ETAD1=ETAD1*RADI*DTOR
      IF(MASH .EQ. 1)ETAD2=CSCRAD*ALOG(1.+ETAD2/CSCRAD)
      DO 45 I=1,NSTRN
```

```
      ETAG1(I)=ETAG1(I)*RADI*DTOR
      IF(MASH .EQ. 1)ETAG2(I)=CSCRAD*ALOG(1.+ETAG2(I)/CSCRAD)
   45 CONTINUE
   50 RETURN
C
  100 FORMAT(4E10.4,I5)
      END
```